PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

Ministry of Higher Education and Scientific Research

University of Mohamed Khider – Biskra

Faculty of Exact Science, Science of Nature and Life

Computer Science Department

Ordre N: IA12/M2/2022

Thesis

Submitted in Fulfilment of the Requirements for the Master Degree in

Computer Science

Option: **Artificial Intelligence**

A deep learning and code of conduct approach for chatbot :

Airline travel information system a case study

By :

GOUICEM Rofaida Ouissem Marthe

Members of the jury :

OUAAR Hanane	MCB	President
MERIZIG Abdelhak	MCB	Supervisor
BELAALA Abir	MAB	Member

2021/2022

Dedication

In the Name of Allah, the Most Gracious, the Most Merciful

I dedicate this work to:

My beloved parents, friends and teachers.

To my parents whom I appreciate their presence, encouragement and charitable advice that keep me motivated to accomplish this work.

To my sisters Amel and Yasmina for their support and my brothers specially Zaki and to my little nephews who have been constantly by my side whenever I felt distressed

To my friends, those with whom I had the most enjoyable moments in my life. To those who affectionately prayed for me and gave me support whenever I got down.

Acknowledgements

First and for most, all praise is due to Allah who gave us the strength to accomplish this work.

I would like to express my sincere thanks to my supervisorDr.**ABDELHAK MERIZIG** who accepted to direct me doing this work, his worthy advice, suggestions, and patience till i finish the work, prompting and fostering us to persist.

I would also to give special thanks **Dr SMAHI Abdeslem** for his dedication and for having responded to all qustions and for supporting me through my work, God bless him .

Our special thanks go to ImadHamzaoui, and khawlaBERHAILI for their helpful suggestion and appreciable directing and support all the time.

I extend my appreciation to my dear friends AmaniKelala, HasnaHadoud,Zomorda,Afaf ,Imad Tibermacine ,Youness Sennousi , Doudou, Amani,Soraya and to my fiancé for believing in me and for encouraging me to accomplish this work .Having such great people around me had a huge impact on my life and pushed me forward to always give my best.

Abstract

In terms of technical advancement, chatbots are a physical manifestation of this human-machine interaction, attempting to comprehend human language as effectively as possible by gathering all available information and data to enhance its potential. In this work, we propose a chatbot system that can interact with the traveller. Combining deep learning algorithms and NLP techniques to predict passengers' intentions based on the question or request they ask, the objective provides a useful answer.

Key words: *chatbot, artificial intelligence, travel assistance, deep learning, natural language processing*

Résumé

En termes d'avancement technique, les chatbots sont une manifestation physique de cette interaction homme-machine, tentant de comprendre le langage humain aussi efficacement que possible en rassemblant toutes les informations et données disponibles pour améliorer son potentiel. Dans ce travail, nous proposons un système de chatbot qui peut interagir avec le voyageur. Combinant des algorithmes d'apprentissage en profondeur et des techniques de NLP pour prédire les intentions des passagers en fonction de la question ou de la demande qu'ils posent, l'objectif fournit une réponse utile.

Les mots clé : *chatbots, intelligence artificielle, assistancevoyage , apprentissage profound, traitement du language naturelle* .

List of Figures

List of Source Code

List of Acronyms

AI Artificial Intelligence

CCE Categorical Cross Entropy

DL Deep Learning

LSTM Long Shot Term Memory

ML Machine Learning

NLP Natural Language Processing

NLU Natural Language Understanding

NLG Natural Language Generation

NER Named Entity Recognition

RNN Recurrent Neural Networks

List of Tables

Content

GENERAL

INTRODUCTION

General Context

Over the past decade, artificial intelligence (AI) has seen rapid growth in adoption in various fields such as automotive, telecommunications, aerospace and healthcare. Among other things the adoption of AI in the travel industry has been slow, the potential value add is high.

With the current world development, it has become even clearer that the travel industry is highly dependent on the changing environment as humans have adopted new habits and travel is highly dependent on technological solutions.

In the domain of travel, artificial intelligence systems offer a number of potential uses. AI helps consumers locate better and more relevant information, offers them more mobility, enhances their decision-making, and ultimately delivers a better travel experience. This is the viewpoint of the customer. Technologies such as Chatbots may be created to serve and support customers of air travel firms as well as the company themselves.

The amount of people who use instant messaging services like WhatsApp, Viber, and Messenger keeps rising today. New insights and possibilities have been unlocked by chatbots for business. Although bot technology has been around for decades, because of the increasing attention of the large companies in Silicon Valley (Google, Facebook, etc.), Machine Learning has progressed substantially, making these technologies both technically and financially accessible...

Problematic and Objectives

When planning a vacation, the majority of individuals do an internet search on websites. The user is confronted with several search engines on the internet, where they look for a spot to submit their trip information, then the website compiles a list of available possibilities. This whole procedure may be rather time-consuming and exhausting, particularly when you have to go through many windows to finish the request.

The global spread of the COVID-19 pandemic has led to a number of limitations and frequent updates to existing and new legislation, posing serious difficulties for the travel industry and many others.

As was mentioned before, the solution to the problem is to use chatbots to make it easier for customers to acquire the information they need, while also enabling the airline or travel firm to supply the information instantly by collecting the data received . In a series of textual exchanges between human and machine, it should be able to respond to various trip requests from several users and assist them through all the trip booking procedures.

The Outlines

Our work is organized as follows:

In the 1st chapter, we discuss the Chatbot technology literature review. To begin, we will discuss the Human Computer Interaction topic, as well as case-based reasoning and its limitations. The problematic description was presented after we had finished defining the chatbot, its types, and its general architecture.Finally, we discussed some related works about the chatbot projects and the methods used.

In the 2nd chapter, we will cover the research methodologies and technologies of chatbots and deep learning. To begin, we will define Natural Language Processing and its history before moving on to analyse its approaches and a definition of code of conduct and its purpose and its efficiency .

The 3rd chapter focuses mostly on Design and Contribution. We describe our proposed architecture and approaches for travel assistance chatbots. The suggested architecture is described with a flowchart and UML diagram, as well as the deep learning method used. Results and implementation are described.

In the 4th chapter, we describe the development tools and frameworks, the chatbot response process, and some screenshots of the program. The findings that were collected are then discussed and analyzed. In the end, we will complete our discussion with a conclusion and our perspectives.

CHAPTER I

CHAPTER I : LITERATURE REVIEW

Chapter 1

Chatbot:Literature Review

1.1 Introduction

Human communication has long been one of the most studied specializations due to its complexity and allure. It has always been fascinating to investigate human language, regardless of the language.

In recent decades, particularly with the advancement of technology, a new communication discipline known as communication or machine-human interaction has emerged.

Artificial intelligence has played a very important role in these technological HMI has quickly become the focus of extensive academic study. A large number of engineers are working on it to enhance and simplify it. They have made progress after many years of work in creating machines that communicate at the same level as humans advance. Natural language processing is one of the most important applications of AI.

Despite its complexity, it has permitted machines to comprehend and simplify human language. Conversational agents, also known as intelligent robots or simply chatbot systems, have changed the world of technology in so many ways that it has been dubbed a "chatbot tsunami." Grudin and Jacques [2019]. The emergence of these intelligent systems, as well as their flexibility, has facilitated their incorporation into websites and mobile applications. These not only focus on detecting and understanding the human language, but also provide a variety of interaction options, such as the ability to order food, buy items, or book a flight, all this without having to use any buttons or menus, simply by asking the chatbot to perform this task.

1.2 Human machine interaction

The improvement of modern data advances such as the Web of things, cloud computing, huge information, inaccessible detecting telemetry, and geographic data frameworks driven to the growth of the investigation of urban insights. Human-computer Interaction could be a broadly inquired about field and examined with a long history as a front-end application innovation in urban insights Qi et al. [2019].

CHAPTER I : LITERATURE REVIEW

1.2.1 Definition

"Human machine interaction (HMI) is an interactive way of information exchange between humans and machines. By collecting the information that can be conveyed by the person to express the intention, and then transforming and processing the information, the machine can work according to the intention of the person." Ren, F., &Bao, Y. (2019)

Human-machine interaction can be referred to as The study of human-computer interaction. This phrase can technically refer to any machine with which people must interact. Among engineers, this concept is known by several other names, including human-computer interface and interfacing. The automatic representation of the concept of Human-Computer Interaction/Interfacing (HCI) began with the advent of the computer, or more broadly, the machine. To improve performance in terms of service quality optimization, a certain level of adequacy must be established between the user, the machine, and the services required, as determined by the HMI. This determines the design's quality in relation to the context in all subjectivity.

1.2.2 Types of Human machine interaction

In today's world, human-machine interactions are diverse and varied depending on the form of interaction.

1.2.2.1 Textual Input

This type of interaction is typically accomplished through the use of Optical Character Recognition (OCR), which is primarily used for the electronic or mechanical transformation of printed, handwritten, or typed characters into machine-encoded text. By recognizing the characters in a picture or scanned document, the process generates a text document that can be edited and manipulated by the computer program Manoharan [2019].

1.2.2.2 Voice based interaction

Speech interface is the new era of conversational agents which supports the interaction between human and machine taking turns in a dialogue.

With voice conversations, the interaction is similar to a conversation between two human users. It's a useful tool for research, for reminders, and for writing notes just by speaking it to the chatbot via natural language Murad, C et al[2018].

1.2.2.3 Facial Expression

“Expressions are a fundamental way to express human emotions and an effective method of non-verbal communication” Shuai-Shi Liu et al[2009].

Facial expression is an interaction mode that uses deep learning to track a user's facial expressions from the camera of an intelligent machine and the algorithms used in this type of interaction learn to understand the facial expressions and execute them.

1.2.3 Case-based reasoning

Case-Based Reasoning (CBR) is a knowledge acquisition and problem-solving technique that is sometimes labelled as Machine Learning.

It's also sometimes linked to other technologies like analogy, cognitive psychology modelling, machine learning, and artificial intelligence. retrieval of information.CBR (Case-Based Reasoning) is an artificial intelligence (AI) paradigm that makes use of prior knowledge to address new problems using information gathered in previous circumstances, referred to as cases. Plaza and Aamodt offered a CBR overview, including a description of the CBR cycle and the key CBR problem-solving techniques Retrieval, Reuse, Revision, and Retention are the four processes. The retrieval method is used to solve a new problem ‹scans the case database for the most comparable cases and selects them using a similarity criterion. the instance with the most repurposing potential. The reuse procedure tries to employ the case that was chosen.The proposed solution is evaluated by the revision process, and if the retrieved case has effectively solved the problem, the problem and proposed solution are saved as a new case by the retention processHomem [2020].Figure1.1 shows the cycle of case based reasoning .

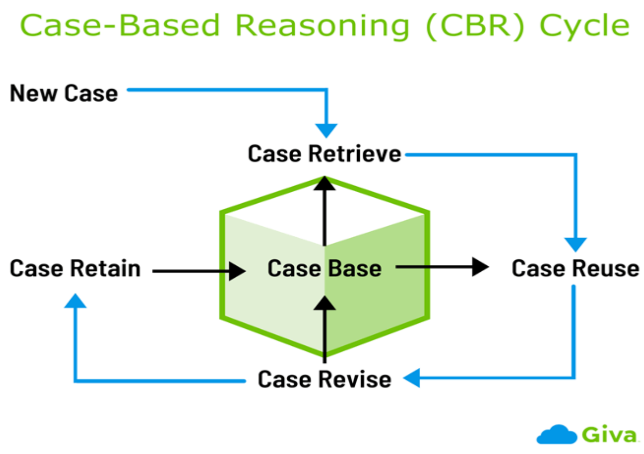

Figure 1.1 The Case-Based Reasoning Cyclefrom Giva [2021].

1.2.4 Limits

1.2.4.1 limitations of case-based reasoning

Since space specialists are inquired to supply information within the frame of specific circumstances instep within the frame of or maybe common IF-THEN articulations, the information procurement handle can be exceptionally quick. The plausibility to consequently upgrade the case base amid the induction to speed up the improvement and support of the utilised information.Comparable programmed adjustment of the knowledge base amid utilisation of an RBR framework is by and large not conceivable. The most issues related to CBR are the requirement for complex remove (closeness) measures and the need for quick get to case bases that some of the time can be exceptionally huge Berka [2020].

Within the CBR framework, there are no strategies that suit each space application, so there are several shortcomings and limits in this frameworks:

•CBR's lack of adaptability is one of its current limitations. In fact, these tools and systems are often built to address a specific issue and hence are not intended to be adjusted by the user depending on the intended use Bentaiba-Lagrid et al[2020].

• In the face of a large knowledge base, CBR is powerless to provide quick access to the case.

• Case based reasoning finds it difficult to deal with natural language which requires a representation of the knowledge, in tight relation with the syntax and the context. Liu,W. [2020].

1.3 New Era of Human Machine Interaction

Due to the rapid rise of technical tools recognized today, engineers and graphic designers are working to build and enhance the interface between human and machine in order to provide a better, more natural, and simple communication experience. Artificial Intelligence (AI) is gradually integrating itself into our daily lives through the development and analysis of intelligent software and hardware, referred to as intelligent agents. Intelligent agents may perform a wide range of jobs, from simple labor to complex procedures. Adamopoulou and Moussiades [2020] describe a chatbot as a conventional AI system and one of the most basic and widely used forms of intelligent Human-Computer Interaction (HCI)

1.3.1 Chatbot

This section of the chapter is a presentation of a novel notion of communication between humans and machines, known as chatbot technology.

1.3.1.1 Definition

A chatbot is a computer program that imitates conversational communication using text or voice Sianaki [2019].

It is a computer software that, when conversed with via text or voice, replies as if it were a clever entity that understands one or more human languages using Natural Language Processing (NLP). A chatbot is described as "a computer software meant to simulate interaction with human users, particularly over the Internet," according to the dictionary. Smart bots, interactive agents, digital assistants, and intelligent conversation entities are all terms used to describe chatbotsAdamopoulou and Moussiades [2020].

A Chatbot (or Chatterbot) is a piece of software (machine) that converses with a human: it's a virtual assistant that can answer a variety of inquiries and provide accurate responses. Chatbots have been increasingly popular in a variety of industries in recent years, including health care, marketing, education, support systems, cultural heritage, entertainment, and many others. Apple Siri, Microsoft Cortana, Facebook M, and IBM Watson are just a few of the well-known Chatbots produced by major corporations for both industrial and research purposes. These are only a few of the most often used systems Saverioet. al[2018] .

1.3.1.2 Brief History

ELIZA is one of the most well-known chatbots. It was designed by Joseph Weizenbaum in 1966 and simulates conversation using pattern matching and replacement methods. Kenneth Colby, an American psychiatrist, created PARRY a few years later, in 1972. The program imitated a schizophrenia sufferer. It makes an attempt to imitate the sickness. It's a natural language program that mimics human thought patterns. Jabberwacky, a chatbot that was founded in 1988 and led to subsequent technological advancements, was another chatbot that arose. Its goal was to create an interesting simulation of a natural human dialogue. Since its inception, some people have used its webpage for academic research purposes. In 1995, A.L.I.C.E. was a universal language processing chatbot that carried out talks using heuristic pattern matching. ALICE was built by Richard Wallace, who was a pioneer in the field. It was previously known as Alicebot since it was the first program to run on an Alice computer.

1.3.2 Types of chatbot

Since the beginning 2000 until today Several chatbots were created, one for each goal to complete and for which it was trained such as Smarterchild (2001) , SIRI (2010) , ALEXA (2014).

1.3.2.1 Voice Bots

The Voicebot is an intelligent voice robot that employs voice synthesis to hear and interpret human speech using natural language processing (NLP) and machine learning methods. The ability to comprehend the meaning and purpose of a user's speech is greatly aided by natural language understanding (NLU). We also refer to this as "conversational IVR" technology. These conversational robots can interpret human speech and comprehend user intent without any additional software Klaus et al [2022].

1.3.2.2 Rule-BasedChatbot

These rule-based chatbots can handle simple queries while aiming to handle complex ones. It works according to a set of rules predefined by a developer using AIML(Artificial Intelligence Markup Language). AIML is a language based on XML

Composer rules for various situations, it is difficult to write rules for every possible situation Thorat et al.[2020].

1.3.2.3 Menu/Button-basedchatbot

"The Menu or Buttons type of chatbot is the one that provides the least user experience and is easiest to build. The following type provides predefined options for the user to select from. The Chatbot can be considered as a glorified decision tree as once the particular option is selected the next preceding options are popped up for the user and will continue until the options reach the leaves of the tree or the user finds the appropriate information in between the nodes of the tree ." Thakkar et al.[2021]

1.3.3 Architecture.

Artificial Intelligence (AI)-based next-generation chatbots require pre-processing techniques such as (natural language processing, language generation, etc...). Ahmad et al.

To comprehend the chatbot's goal and categorization, developers must choose the appropriate algorithms, platforms, and tools. To avoid problems, developers build the chatbot in chunks. This aids end users in their planning Stephen Roller [2020].

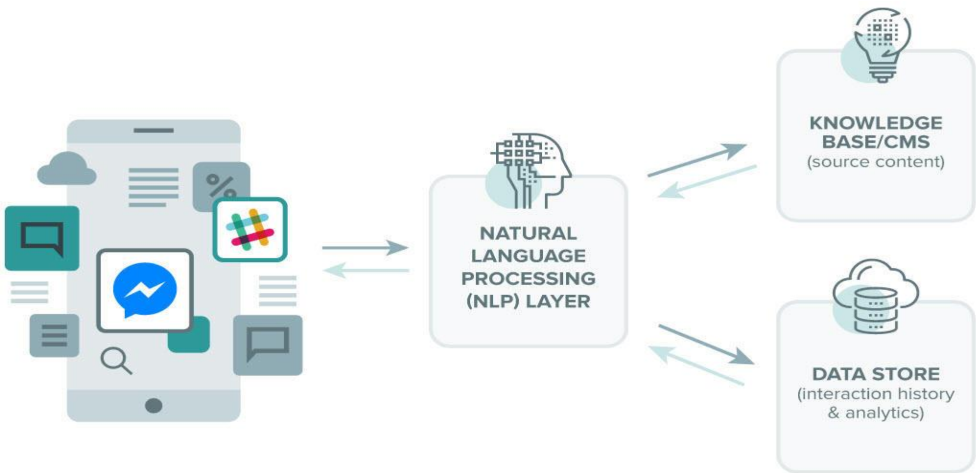

Figure 1.2 General Architecture of ChatbotSubramanian [2018].

1. User interface is where the system gets the input of the user
2. Natural language processing (NLP) allows chatbots to transform text and voice into machine-understandable data.
3. A knowledge base is a collection of data from which a chatbot may get information for the purpose of responding to users.

4. Data Store is where developers may preserve chatbot chats for customer service, training, and testing (SQL , cloud etc..)

1.4 ProblemStatement : Air Travel Assistance

The usual method of booking a flight or other mode of transportation involves visiting a location or using a website to do so. This section illustrates how our challenge may be used to help travellers.

1.4.1 Definition

Most traditional booking methods, which require the passenger to go to the airline office to make a reservation or any other operation, expose him to a number of problems, in addition to the consequences of covid-19 with the health protocol.

It is almost impossible for a single office to provide all the information that a number of customers need. Even if the company has many representatives, the problem remains.

In order to solve these problems, major airlines have resorted to technology through email and websites; however, these do not provide the expected information due to lack of clarity, saturation of websites due to high demand, or the time it takes for company employees to respond to emails.

1.4.2 Benefits of the Chatbot in Air-Travel Assistance

The use of chatbots in the travel industry carries with it a number of advantages.

- There is a financial benefit for airenne companies by reducing the number of employees who only work a certain number of hours per week and take breaks, not to mention health concerns, while the chatbot responds twenty-four hours a day, seven days a week, without taking a break, and with a reliability that makes airenne companies adopt it Thazhathethil et al [2021]
- The chatbot is much cheaper than the development of some software Palanica[2019].
- Fast access to the information that is being sought Chatbots make it possible for consumers to communicate with airlines at any time, eliminating the need for them to consider factors such as different time zones, business hours, and waiting lines at contact centres Zumstein et al [2017].
- Automatic data and information collection takes place with every interaction that takes place between the chatbot and the user (purchase history, problems,

etc.). This data gives the Aireen firm the ability to update their systems and so enhance their client service. Additionally, they may now update their data. The user now has access to the history of the chatbot, which enables him to refer to his request whenever he sees fit. This upgrade was made possible thanks to the previously mentioned changeChrysovelidis [2019].

1.5 Related Works

Chinedu et al.[2021] developed an online educational platform where students can learn using Chatbottechnology.Thesechatbots are considered as a useful technology to facilitate learning within the educational context. Chatbots are conversational or interactive agents that provide instant response to the user. This work was based on the title, abstract, and keywords (TITLE ABS- KEY) the defined search words were combined in various ways to search for the appropriate articles in the selected databases.

Authors in Eric et al.[2021] proposed a KB design framework that incorporates customer knowledge management (CKM) .The authors presented a system architecture of a chatbot that proactively improves itself continuously which is built on the customer knowledge management process . The chatbot conversation deals with different types of customer knowledge. In order to develop a system based on the proposed design and piloted and evaluated in a multinational intimate apparel company.

Tran et al.[2021] investigates the differences in consumers' sentiments towards chatbots across retail sectors, and the influence chatbots have on consumers' sentiments and expectations towards other service interactions with online human agents. Only the fashion and telecommunications sectors were studied in this study.Theyanalyzed consumer sentiment towards chatbots and online human agents on Twitter through a hybrid model combining a VADER lexicon-based classifier and Naïve Bayes traditional machine learning model (Kübleret al., 2019).Authors used an open application programming interface (Rapin and Dunn, 2003) provided by Twitter to gain access to the platform data.They gathered 8190 raw tweets using this way.According to the findings of this study, consumers feel more positively about their interactions with chatbots than they do with online human agents.

1.5.1 Synthesis

This section compares the chapter's works. This comparison is based on five parameters: topic, purpose of the work, platform, chatbot type, language used for each research, and methodologies used, which are the primary focus of our work due to their relevance in decision-making.

	Chineduet al.[2021]	Tran et al.[2021]	Eric et al.[2021]	Our work
Domain	Education	Retailsectors	Commerce	Air Travel
Objective	learning	Investigate differences in consumer experience with chatbots versus online human agents and the impact of chatbot implementation on consumer expectations	propose an intelligent knowledge-based conversational agent system architecture to support customer services in e-commerce sales and marketing.	Air Travel assistance
platform	Online platform	Application	Framework	Mobile Application
Chatbot type	Conversationalchatbot	Conversationalchatbot	Conversational agent	Conversationalchatbot
Method	Title Abd-key (AND , OR process)	Social media data, sentiment analysis	BILOU , DIET, spaCy,FastText, FAISS.	LSTM
Language	English	English	English	English ,arabic , French

Table 1.1: Relatedworkcomparison

1.6 Conclusion

Chatbots are the new era of human machine interaction as mentioned above. case based reasoning has become hard to implement nowadays especially with the development of techniques of AI . Next chapter we will discuss the development of chatbots using NLP methods and deep learning algorithms.

CHAPTER II

CHAPTER II : NATURAL LANGUAGE PROCESSING

Chapter 2:

Research methods : Natural language processing

2.1 Introduction

Much has been written about communication, specifically the challenges of speaking across age, gender, exceptions, ethnicity, or educational divides, and the need of maintaining and developing human relationships. Artificial intelligence (AI) research has led to the development of software and robots that have the same range of abilities as humans. The ability to communicate with artificial entities has become a need. Much has been written about communication recently, natural language processing has achieved popularity for being an effective method which represents, and analyses human discourse computationally. A variety of domains had led to the growth of its application such as email spam detection, information extraction, summarization, medical and question answering, among others.

2.2 Definition

There is no exact definition of natural language processing (NLP). According to Y. Goldberg [2017] Natural Language Processing (NLP) is a set of theoretically motivated computer methods for analysing and designing texts that appear naturally at one or more levels of linguistic analysis to obtain human-like language processing for a range of tasks and applications. As AlpaReshamwalaet al.[2013] define it as a group of methods which derive grammatical morphology and meaning from input in order to accomplish an appropriate activity. As a result, the target language's rules and the task at hand determine natural language generation output creation. NLP benefits training systems, duplicate detection, computer supported coaching, and database interfaces because it allows for more engagement and productivity. Online information retrieval, aggregation, and question-answering are examples of NLP which have been mainly based on algorithms relying on the textual representation of web pages K. R. Chowdhary [2020].

CHAPTER II : NATURAL LANGUAGE PROCESSING

NLP is an integrative field that combines computational linguistics, computing science, cognitive science, and artificial intelligence. It considers the use of computers to process or interpret human (natural) languages in order to perform meaningful tasks Li Deng et al.[2018].

Its main objective is to create systems that are capable of understanding text and performing tasks like translation, spell checking, and topic classification automatically.

2.2.1 Techniques to train NLP model

Machine learning is an artificial intelligence application which allows a system to learn from previous experiences automatically. The system does not need to be explicitly programmed. Machine learning can be used in synthesising the basic relationships across a wide range of data sets. This data set is used to solve real-time challenges including big data analytics, information evolution, and deep learning which are subset of machine learning. Deep learning processes can convert numerous features. Thus, it is preferred to compute massive datasets and unstructured data. Deep learning makes it easier for computers to analyse and extract crucial information from a large amount of data. Both of these strategies have allowed computers to solve complex issues like speech recognition, object identification and perception BhaveshSingh et al [2021] .

As it is illustrated in the figure below, natural language processing is a subset of artificial intelligence (AI). It is involved with the relationship between human and computer languages. Computers can analyse natural language in the form of speech and text using natural language processing. Machine learning design has been used to understand text content for a long time and has demonstrated to be effective. The language must be converted to numerical form, since machine learning algorithms can process numerical data only. On the other hand, deep learning has enabled computers to create ways to convert text data to numerical form in order to develop more precise representations of data and characteristics for complex tasks Bhavesh et al [2021]. Figure 2.1 shows the relation between all thesetechniques .

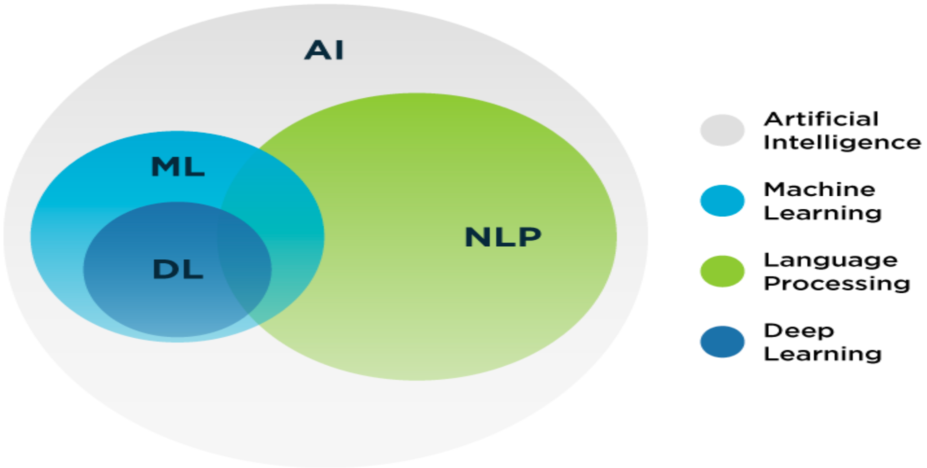

Figure 2.1 Representations of AI, ML, DL and NLP from Kaviraju [2021].

2.2.2 Natural LanguageUnderstanding (NLU)

The meaning of a sentence is determined by natural language understanding via syntactic and semantic analysis of text and speech. Syntax refers to the grammatical structure of a sentence. However, semantics refers to the intended meaning. NLU also set up a relevant principle, which is a data structure that defines the relationships between words and sentences. While people do this naturally in conversation, a machine will need to integrate these analyses in order to understand the intended meaning of varied texts. Artificial intelligence (AI) is used by NLU to understand information received in the form of text or speech. It attempts to figure out the meaning of the written text. After speech recognition software transforms words to text, NLU software decodes the meaning, even if they include common human mistakes or mispronouncing. Young et al [2020].

It is absolutely possible for the same text to have various meanings, that different words can have the same meaning, or that the meaning differs depending on the context.

The following three strategies, however, are used to comprehend natural language:

-Syntax: comprehends the text's grammar.

-Semantics: is the study of the text's actual meaning.

-Pragmatics: comprehending what the text is attempting to convey, in other words the context.

2.2.3 Natural LanguageGeneration (NLG)

Natural language generation (NLG) is a process for transforming raw structured data into plain English or any other required language. Data storytelling is another term for this. Many firms that employ a lot of data benefit from this strategy. It transforms structured data into natural language in order to effectively communicate patterns or detailed insights in any domain Fadhlallah [2021].

2.3 Brief History

Natural language processing research has been continued since the late 1940s, following WWII. Machine translation (MT) was the first computer-based translation system. People realised the importance of translation from one language to another at this time, and they hoped to develop a machine that could do it automatically. However, the task was clearly not as simple as people had hoped Khurana et al. [2017].

NLP researchers were divided into two groups from 1957 to 1970: symbolic and stochastic. Symbolic, or rule-based, researchers focused on formal languages and syntax generation, seeing this as the start of artificial intelligence research. Chomsky (1957) was the first to introduce the concepts of a finite-state machine to linguistics, as well as context-free grammar (CFG) LISP (Locator/Identifier Separation Protocol) was created by John McCarthy in 1958 and is still used today. The first expert system was developed in 1970, with the main algorithm being inference rules in the form of "if-then-else." Deng and Liu [2018]

CHAPTER II : NATURAL LANGUAGE PROCESSING

In the early 1980s computational grammar theory, which is linked to logic for meaning and knowledge's ability to deal with users' beliefs and intents, as well as functions like emphasis and themes, became a forceful area of research Johri et al. [2020].

Discriminative models have been the de facto strategy in a range of NLP problems since the late 1990s.

During the second NLP wave, empiricist machine learning and linguistic data analysis began in the early 1990s by crypto-analysts and computer scientists working on natural language sources with limited vocabulary and application areas. During this time, building deep neural networks from the ground up drew a lot of attention. One of the first human–machine corpora is the Air Travel Information System (ATIS) Pilot Corpus Hemphill et al. 1990) Deng and Liu [2018].

Much work since around the year 2000 has involved the use of machine learning techniques such as Bayesian models and maximum entropy. This has involved using annotated corpora to train systems to segment and annotate texts according to various morphological, syntactic or semantic criteria. These techniques have been systematically applied to particular tasks such as parsing, word sense disambiguation, question answering and summarisation. In the year 2011, Apple'sSiri became known as one of the world's first successful NLP/AI assistants to be used by general consumers R. Kibble [2013].

2.4 NLP Techniques

2.4.1 Tokenization

Tokenization is one of the first steps in any NLP pipeline. It is the process of breaking down words into smaller, more meaningful parts called tokens and deleting non – essential elements like punctuation. These are also known as the smallest individual units. Words, punctuation marks and special characters are the smallest units of words in human languages. In the form of a type/token distinction, these tokens are commonly referred to as terms or words. A token is a unique instance of a group of letters in a document that have been put together as a meaningful semantic unit for processing Jyotsna K. Mandal et al. [2020].Figure 2.2 shows a Tokenizationexample .

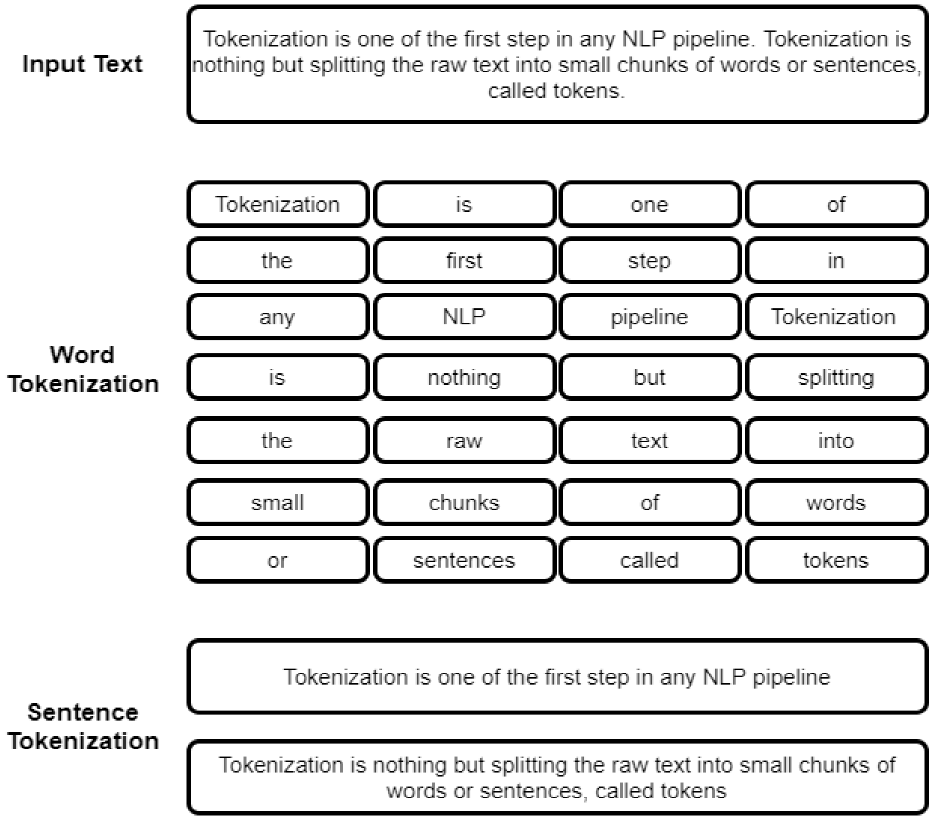

Figure 2.2 Tokenization Method Example.

2.4.2 Text Normalisation (Stemming and Lemmatization)

The stemming or Lemmatization NLP technique seeks to generate root words from these word variations. Stemming is a crude heuristic process that attempts to achieve the mentioned goal by chopping off the ends of words, which may or may not result in a meaningful word in the end. Lemmatization, on the other hand, is a more advanced technique that aims to do job efficiently by utilising a vocabulary and morphological analysis of words. By deleting the inflectional endings, it returns the base or dictionary form of a word termed a lemma. When it comes to boosting the relevance and recall skills of a retrieval system, both stemming and lemmatization play critical roles. Because the system will use one index to provide a number of comparable words with the same root or stem when these strategies are applied, the number of indexes used will be minimised Balakrishnan et al. [2015].

Truncating methods, statistical methods, and mixed approaches are the three types of stemming and lemmatization algorithms. The stems or lemmas of the word variations are obtained in a consistent manner by each of these categories YounessTabii et al. [2018].Figure 2.3 Show comparabilitybetweenstemming and lemmatization .

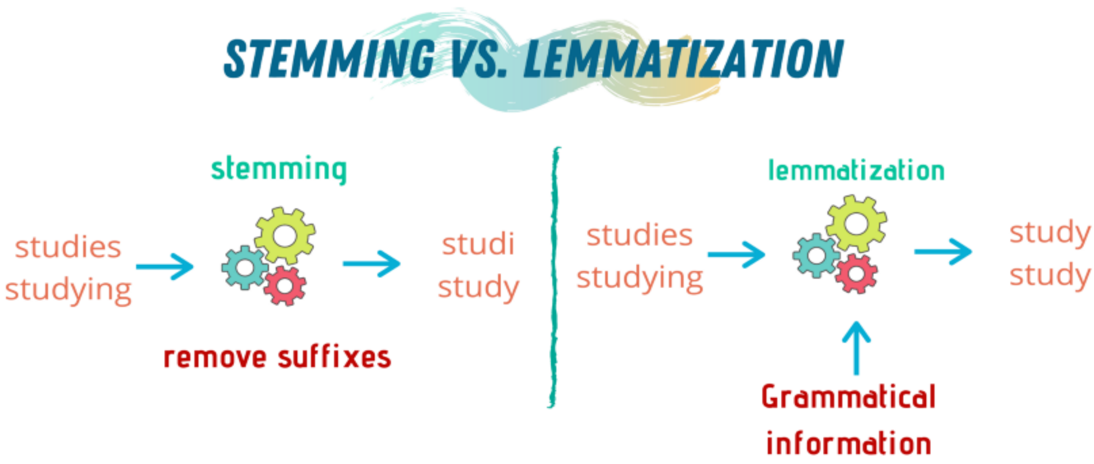

Figure 2.3 Stemming vs. Lemmatization Example from SatishGunjal [2019].

2.4.3 Stop Word Removal

A stop word is a frequently used word (such as "the," "a," "an," or "in") that a search engine has been programmed to ignore, both when indexing and retrieving entries as the result of a search question.

In order to reduce the size of the text and to enhance the performance of the information retrieval system, stop-word removal has been taken as a relevant pre-processing technique used in natural language processing applications. Yet stop word removal is not a definite NLP technique to make for every model as it depends on the task Raulji et al. [2016] . Figure 2.4 shows a stop wordexample .

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

Figure 2.4 Stop Word Example from GeeksforGeeks [18 May, 2022].

2.4.4 NamedEntity Recognition (NER)

It means the task of identifying named entities like person, location, drug, time, organisation, clinical procedure, biological protein, etc…in text. NER systems are frequently used as the first step in question answering, information retrieval, conference resolution, topic modelling, etc. So, it is important to highlight current advances in named entity recognition, especially recent neural NET architectures which have obtained state of the art performance with minimal feature engineering Jing Li et al. [2020].Figure 2.5 shows namedentity recognition example .

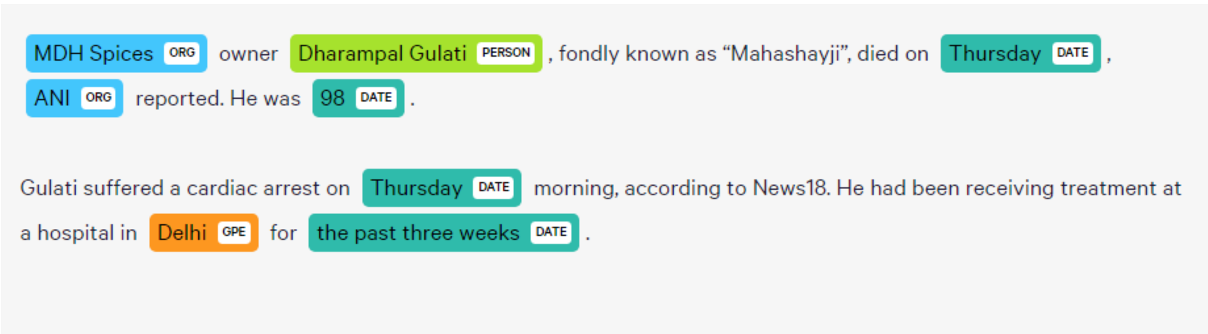

Figure 2.5 Named Entity Recognition Example from CharudattaManwatkar [Dec 12, 2020].

2.4.5 Sentiment Analysis

The process of extracting and comprehending the emotions defined in a written document is known as sentiment analysis. The explosion of data on social media platforms such as Twitter, Facebook, and LinkedIn has offered customers new methods to voice their opinions about products, people, and locations. The user's opinion is always expressed as textual data. Millions of text messages are sent every day via social media and online shopping platforms. Investigating and assessing the thoughts of the public's opinion is a crucial duty. To evaluate if the sentiment of an opinion is good, negative, or neutral, NLP with artificial intelligence capacity and text analytics are applied .Shetty et al. [2017]. Figure 2.6 sentiment analysisexample .

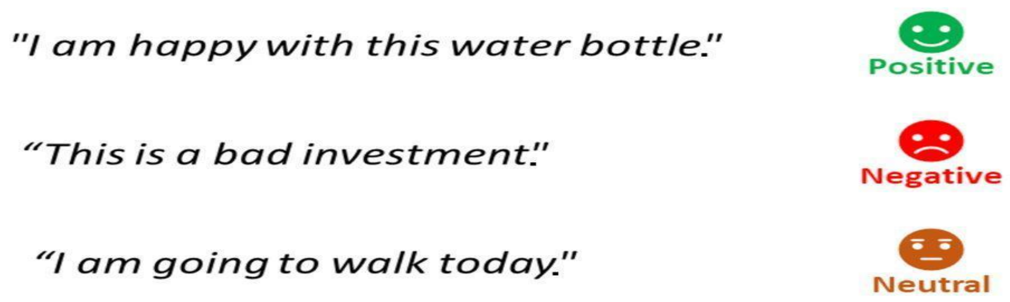

Figure 2.6 Sentiment Analysis Example.

2.4.6 One-Hot Encoding

One-hot encoding is one of the various methods that may be used to encode categorical variables for modelling. However, it is also one of the methods that is used the most often.When the characteristics are nominal, we use this categorical data encoding strategy (do not have any order).During the process of one-hot encoding, a new variable is generated for every level of a categorical feature, and each category is mapped to a binary variable that contains either 0 or 1 .Dahouda et al.[2021]

The one-hot encoding technique is a widespread and fundamental method that transforms a word into a vector. This process results in the production of a binary vector of length N, where each word is assigned its own distinct integer index. P. Cerda et al[2020] . N is equal to the size of the vocabulary. One way to express a word is by using a vector that all of it zeros with the exception of one element, which represents the word itself and has a value of one Dongsheng Wan[2021].Figure 2.7 One-Hot encoding example.

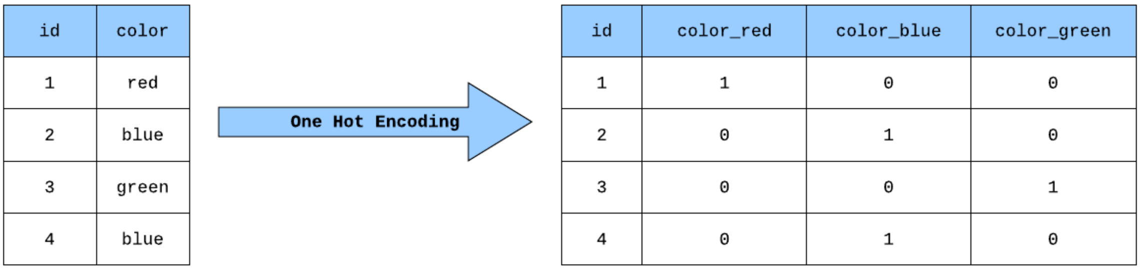

Figure 2.7 One-Hot encoding example from George Novack [2020].

2.4.7 Topic Modelling

For nearly two decades, topic modelling has been an effective text analysis technique. In general, it is regarded as a collection of qualitative approaches because it statistically pre-processes text data, despite the fact that the findings are always qualitatively interpreted. It is also known as an inductive technique with quantitative measurements, which makes it ideal for descriptive and exploratory research.

Different subjects are extracted from an existing text in topic modelling. An overview of the text can be gained by organising them thematically and preparing them for further study in this fashion. By examining the sentiments inside the collected topics, topic modelling approaches are frequently integrated with sentiment analysis.

In recent years, a variety of topics modelling algorithms have been developed, with Latent Dirichlet Allocation (LDA) being the most well-known and commonly used algorithm Egger [2022].Figure 2.8 Topic modelling example .

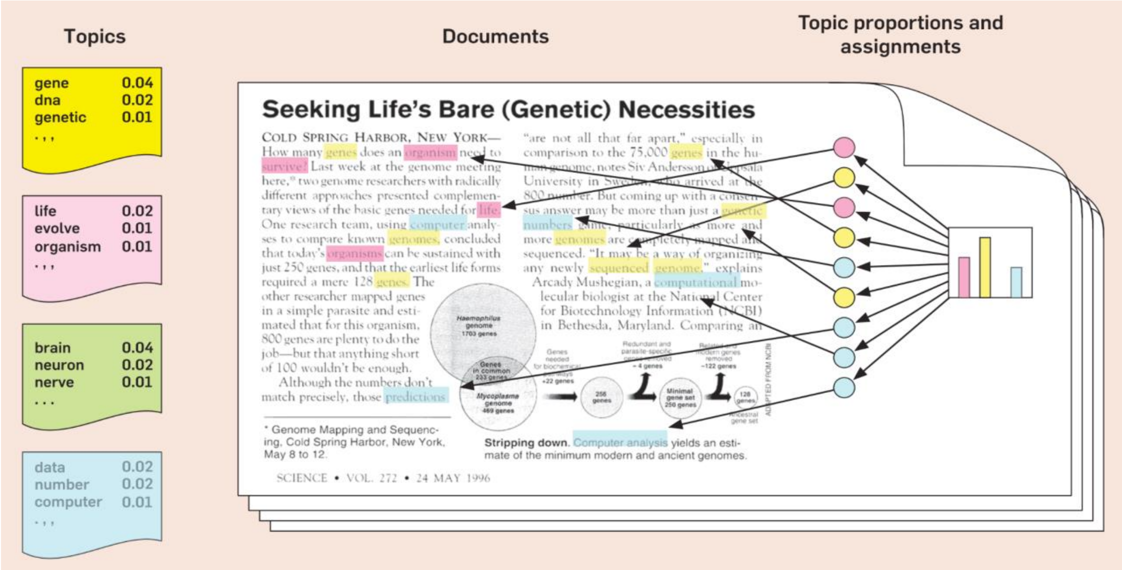

Figure 2.8 Topic Modelling Example from Tarray, T. A et al[2019].

2.4.8 Word of Cloud

Word clouds have remained a common method of summarising textual data. As the big data era approaches, it is defined as a type of weighted list used to visualise language or text data, which is growing in popularity and application potential. Users with basic requirements, such as repeating a specific phrase or extracting text data from a web page, can now use a variety of online word cloud generators. Furthermore, most current word cloud generators do not support characters other than English, limiting users who do not speak English. There are also word cloud-generating tools for programming languages (such as Python and R), but these require coding and are not user-friendly.

The font size and colour tone of each word in this cloud are distinct. As a result, determining prominent terms is aided by this representation. The importance of a word in relation to other words in the cluster is indicated by its increased font size. Depending on the writers' concept, word clouds can be constructed in a range of shapes and sizes. The number of words in a Word Cloud is quite important. More words may not always make for a better Word Cloud, as it can become congested and difficult to read. A Word Cloud should always be semantically meaningful and represent the purpose for which it was created.In the twenty-first century, as internet technology advanced, particularly websites and blogs became more widely used, word clouds were used to place tags on web pages as navigation aids for readers' information by visualising the frequency of each keyword by font size Yuping Jin [2017]. Figure 2.9 word of cloud example .

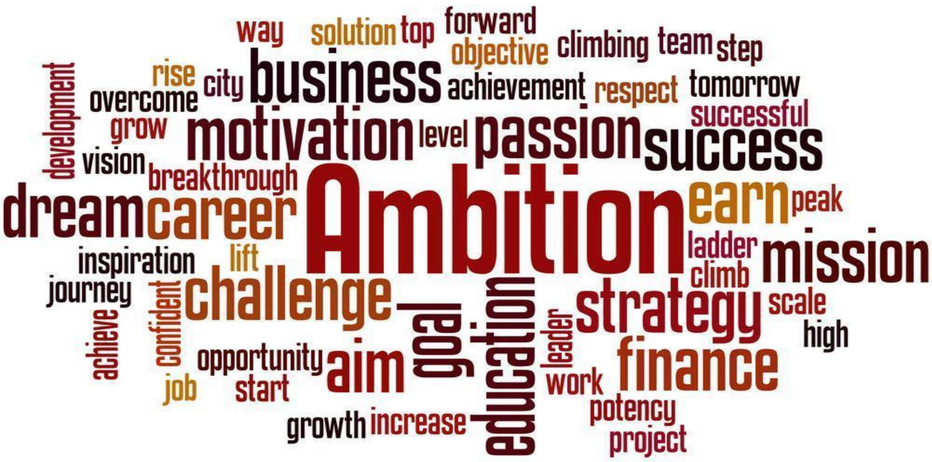

Figure 2.9 Word of Cloud Example from rakus [30 July 2017].

2.5 Code of Conduct

Codes of conduct are guidelines adopted by developers to address what behaviours are expected and appropriate from their platform's users. A code of conduct, also known as privacy and code of conduct, is a defined set of rules, guidelines, values, employee expectations, behaviours, and relationships that a developer needs to take into consideration and believes is required for the success of its final product.This code is based on best practices from other industries and integrates advice from privacy, civil liberties, and consumer advocates, app developers, app publishers, and other organisations across the mobile ecosystem. The transparency created by consistently displaying information about application practices as specified in the code is aimed to aid consumers in comparing and contrasting app data practices with the goal of improving. The brief notices are intended to increase consumer trust in app information practices.This code is meant to reflect existing application practices without discouraging mobile applications to notice innovation or interfering with or undermining the consumer's experience Ntia [2014].

Professional marketers' obligations to respondents and the client organisation are guided by codes of conduct; there is no expectation that they will guide SaQ respondents, despite respondents, who agree to participate in a market research project of their own free will, having at least an implied ethical obligation to act professionally and provide honest and trustful answers while maintaining confidentiality.

Professional codes of conduct provide a template for systematic and ethical decision making. There is a normative tradition within the marketing discipline of developing guidelines or rules to assist marketers in moral decision making and acting ethically (Hunt &Vitell, 1986).

The American Marketing Association, for example, provides a code of conduct as a structure for self-regulation, organising and assessing alternative courses of action, establishing ethical rules for conduct, boosting public confidence, and reducing the need for governmental and/or intergovernmental legislation or regulation among professional marketers Agrawal et al.[2014] .

2.5.1 Purpose of Code of Conduct

The aim of this Code of Conduct (hereafter the 'Code') is to promote confidence among mobile applications users that process personal data, such as health data. Before using an app, health apps must give users clear and outstanding information about how their data will be utilised. This will help ensure that data are used fairly and transparently, which is essential for building trust. As a result, the code strives to make data protection compliance easier and to help best practices in this domain Net et al. [2016].

The Code determines to provide clear and easy- to understand instructions on how to resolve data protection legislation to apps. This piece of advice is addressed to app developers, which are individuals, corporations, or organisations who make software apps for mobile devices available (either directly or through app stores) to handle data in any field Edwards-Stewart et al. [2019].

The Code of Conduct is meant to increase confidence amongst mobile apps users while also giving app developers who have subscribed to it a competitive advantage Ronald Leenes et al. [2017].

2.5.2 Effective Code of Conduct

Determines and defines appropriate and inappropriate behaviours. It goes beyond the ethical management of data to cover the treatment of people, as well as clearly defining reporting and investigating procedures and outlining disciplinary consequences for violations of conduct, as well as protection against retaliation Sengodan et al. [2019].

2.6 Conclusion

In this chapter, we explored natural language processing and its significance in artificial particularly with regard to the development of chatbot systems.The techniques that are employed in NLP provide an exceptionally effective approach to comprehend human language. The primary focus of our work is on developing a deep learning and natural language processing (NLP)-based chatbot that is both effective and fully operational.We'll talk about our system's architecture, deep learning algorithms, and Natural Language Processing techniques in the following chapter.

CHAPTERIII

CHAPTER III : DESIGN AND CONTRIBUTION

Chapter 3

Design and contribution

3.1 Introduction

The methodologies of natural language processing were discussed in the previous chapter. Our goal is to realise an Android mobile chatbot application for air travel information that uses natural language processing methods and deep learning algorithms to fulfill the user's needs and deliver a useful result.

The proposed architecture of our system, as well as the deep learning algorithms and natural language processing approaches required to achieve our goal, will be discussed in this chapter.

3.2 Proposed Architecture

The goal of the research is to create a search-based chatbot model utilising recurrent neural networks. This chatbot's model is made up of three layers, each one serves a separate function. The user interface is the initial layer through which the user expresses his or her request or question in text. The processing layer then deals with information reception and processing, starting with translation if necessary based on the language detected with Google API translator, filtering any xenophobic expressions and offensive terms, and then applying natural language processing methods to the input for classification of intentions and entity extraction. Based on the information gathered, the chatbot response is retrieved from the Data Source layer that contains the Air travel dialogue state tracker answer, the response translated into the user's detected language and outputs it on the user's interface. The Figure 3.1 belowshow’sour system architecture.

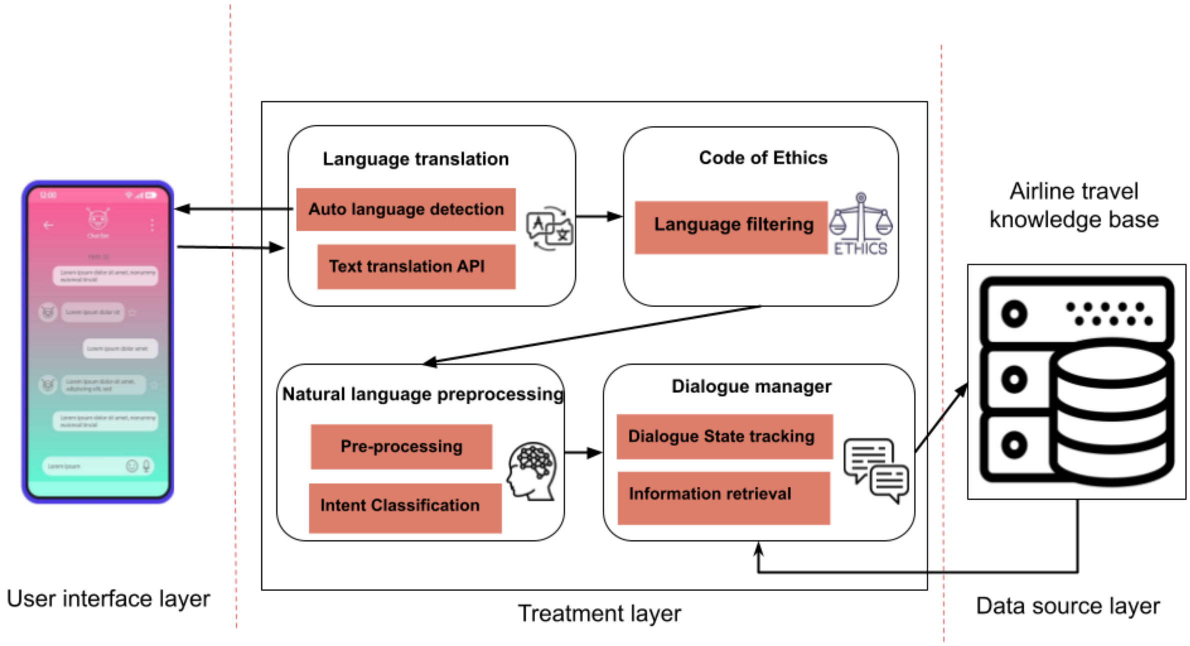

Figure 3.1 : The proposed architecture

3.2.1 Architecture description

This section provides a detailed description of the role of each component seen in figure 3.1 above.

1. Interface layer

The first contact between the user and the chatbot is through the first layer.in which there are two parts the user passes his request or question through an input text bar which is none other than the screen of the android mobile to send it to the Processing layer. The second part is an ordered scan window that contains the entire history of the conversation between the user and the chatbot.

1. Treatment Layer

This layer is made up of four different modules that represent the entire chatbot system because it comprises the majority of the functions and main methods.

a. Language Translation

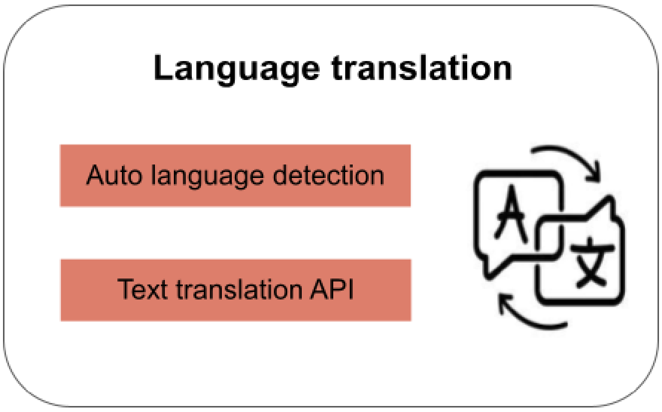

Figure 3.2 Language translation components

As shown in the figure 3.2 the translation between user input and the chatbot system is handled by this module, which employs an artificial language detection mechanism with three supported languages (English - Arabic - French). Requests a translation from the user's detected language to the chatbot system using the Google API translation server. We employ the same technique to convert the chatbot system's response from English to the user's chosen language.

b. Code of Conduct

At the level of this module the message transmitted by the user and after having been translated by the language translation module undergoes a filtration of any xenophobic and offensive expression which can shock the user, the chatbot rejects any request of this kind which are already listed beforehand in the three languages (French, Arabic, English) linked in the previous module

c. Natural languageprocessing

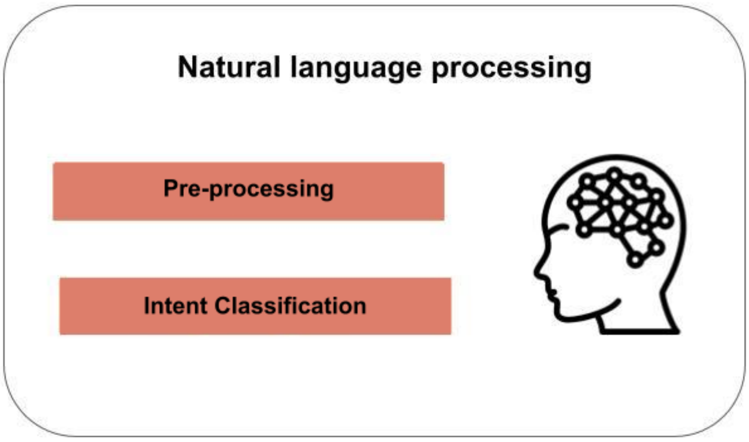

Figure 3.3 Natural language components

This module is responsible for understanding the text received for the user's intention which consists of 3 tasks, pre-processing of the text, extraction of entities and intentionsClassification.

- **Pre-processing :**

Tokenization, stemming, and One-Hot encoding should be applied, along with lowercase letters, clearing, and elimination of unnecessary symbols such as punctuation and stop words.

- **Intent classification :**

Uses the One-Hot encoded sequence in order to apply our model prediction for intent classification.

d. Dialogue manager

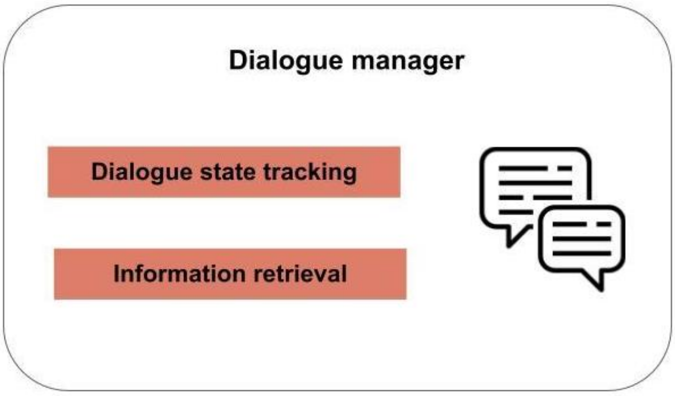

Figure 3.4 Dialogue manager components

- **Dialogue state Tracking**

Dialogue state tracking is a dialog manager component responsible for the flow of the conversation between the user and the chatbot. It keeps a record of the interactions within one conversation in order to decide how to respond.

- **Information Retrieval**

In order to retrieve the chatbot answer from the knowledge base, this task will be sent to the translation API after the dialogue state tracker has been built.

2. Data source Layer

This layer contains a database about Answers and information about the services and topics. The database feeds the chatbot with the information it requires to give a suitable response to the user.

3.2.2 Air Travel Assistance

The following phases, which are given in the below flowchart, are used by the proposed chatbot system in order to answer to the user's request (see Figure 3.6).

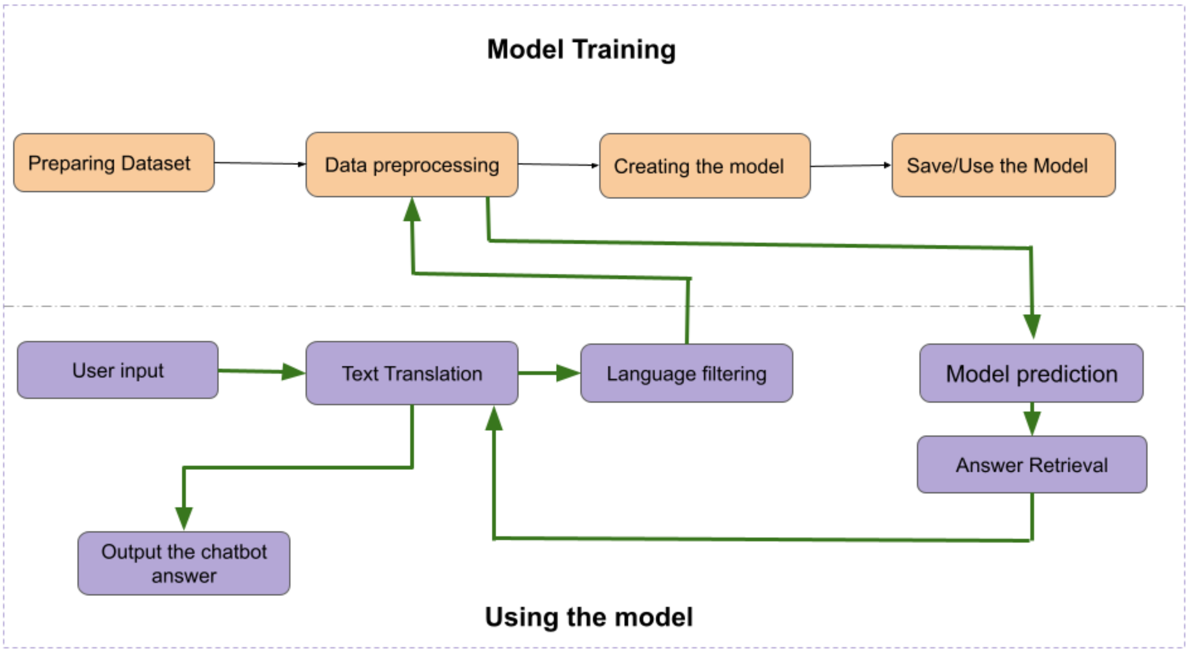

Figure 3.5 Air Travel assistance process

1.Model Training

a. PreparingDataset

A csv file with two columns, Question and Intent, containing the data from numerous files has been assembled and is ready to be fed into a machine learning algorithm.

b. Data Pre-processing

The quality of the data defines the efficiency and credibility of the machine learning model in order to feed the model with data. As a result, it must go through the pre- processing steps.pre-processing methods, lower case, punctuation removal, empty words removal, tokenization, truncation, application of hot coding to our data to obtain the representation with the same length as the vocabulary size (number of unique words) at each sequence.

c. Creating model

The creation of a precise learning model is divided into two stages:

• configuration of some parameters: number of LSTM cells, number of layers, activation functions...etc.

• Training the model and testing it (making predictions to test the model).

d. Save/Use the Model

Following the creation of the model, it must be exported (saved) for subsequent use in model prediction.

2.Using the model

a. User’s input

After loading the text-based user input from the application interface, it will be transferred to the local server for processing using FLASKframework-based python scripts.

b. Text translation :

A python lang detect package is required to determine the language of the user's input. GOOGLE API servers (English, Arabic, and French) will automatically translate any input text that isn't already in English. Languagefilteringreceives the input after translation.

c. Language filtering :

After we have received the text that has been translated, it will be screened to remove any xenophobic or inappropriate language using the Speech Hate API.

d. Data preprocessing :

As soon as the translated user's text input is loaded, we clean and remove any unnecessary symbols, such as punctuation and stop words. The text data was then tokenized and part-of-speech tagged in order to extract the entities. Lowercase, punctuation and stop words are removed from the text before tokenization, stemming and the identification of named entities.

e. Model Prediction :

To put our model's predictions into action, we'll use a technique called One-Hot encoding to turn the text input into numerical sequence values. Dialogue manager will be notified of the identified purpose and the retrieved entity's information.

f. Answerretrieval :

Construct a dialogue state tracker based on the purpose and the intents that were collected; this tracker will be used to obtain the chatbot answer from the knowledge base. After that, it will be sent to the translation department.

g. Outputs the chatbotanswer :

The chatbot response will then be shown in the application's conversation layout after it has received the text input from the user.

3.2.3 UML Diagram

In this section, we detail our chatbot's functionality in which the user can receive responses.Steps and tasks are shown in the Sequence diagram in Figure 3.7 below

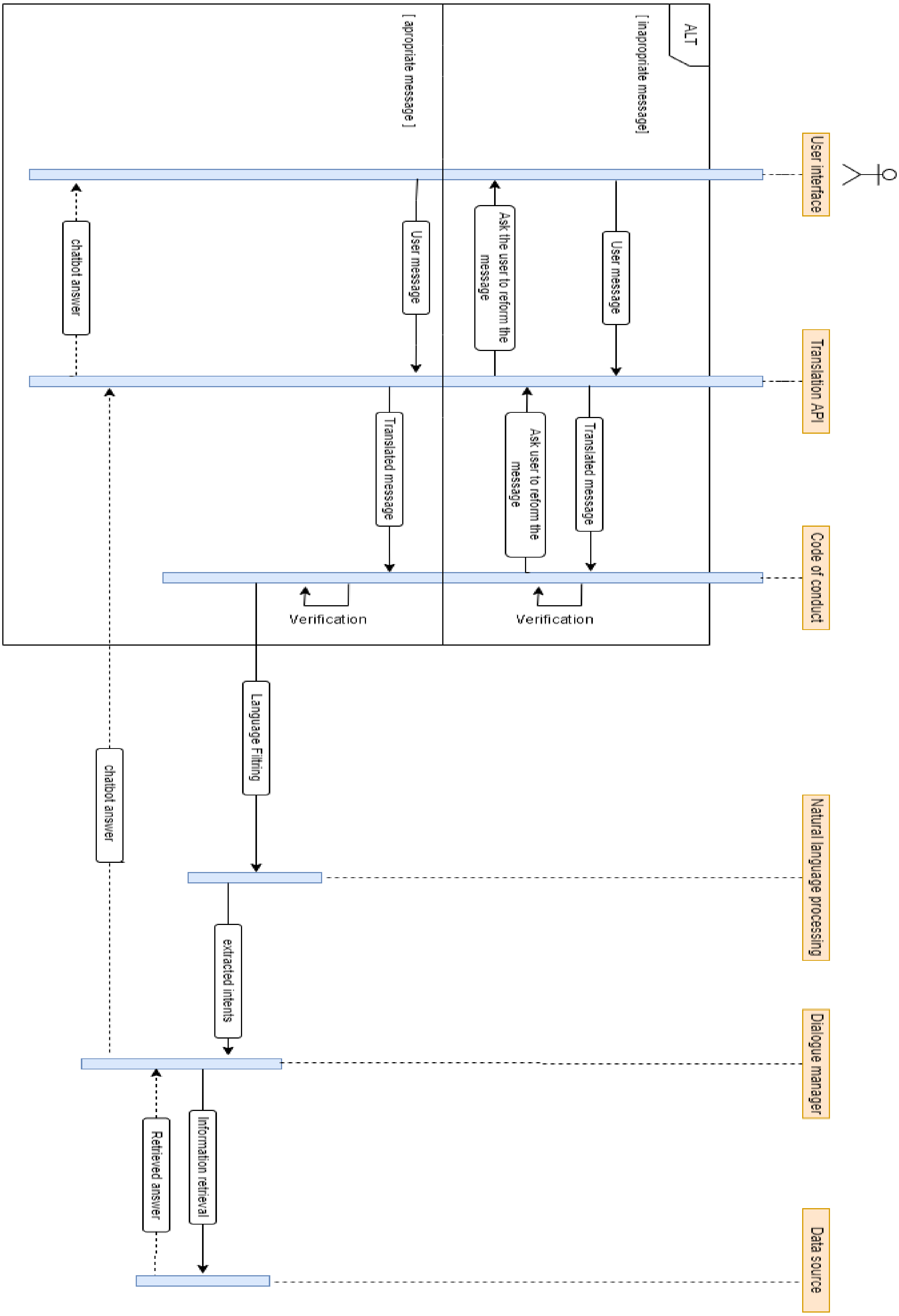

Figure 3.6 The chatbot application's sequence diagram

3.3 UsedAlgorithms

3.3.1Recurrent Neural Networks (RNN)

Research in the fields of sequential data such as text, audio and video is mainly used by recurrent neural networks (RNNs). However, RNNs composed of sigma or tanh cells are unable to learn relevant information from input data when the input deviation is large.Yong Yu et al.[2019]

The RNN has a rather atypical architecture that consists of a cyclic connection; the essential characteristic of an RNN is the memory capacity . The RNN receives inputs and emits outputs. From the pass states and the current input data , these cyclic connections allow the RNN to have the ability to update the current state .

During each update cycle the RNN also exchanges information with the memory matrix via the read and write heads. Manaswi, N.K. [2018].

These networks, such as full RNA and selective RNA composed of standard recurrent cells (e.g. sigma cells), have been incredibly successful for some problems. Figure 3.8 show an RNN architecture

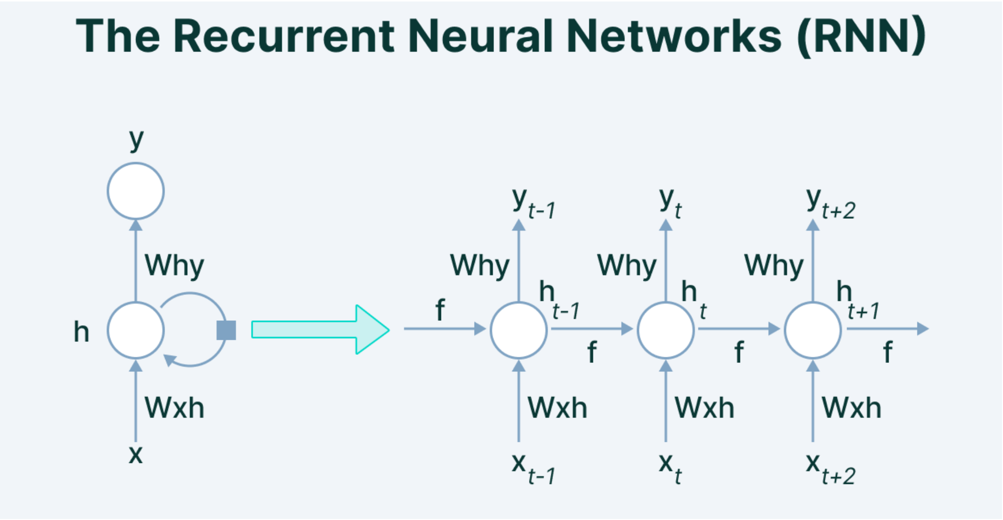

Figure 3.7 Recurrent Neural Networks architecture from PragatiBaheti [May 26, 2022]

3.3.2 Long-Short Term Memory Network (LSMT)

Long short-term memory is a modified RNN architecture that addresses the problems of vanishing and exploding gradients, training over long sequences, and memory retention. Manaswi, N.K. [2018] .The LSTM is a specialised form of RNN that is distinguished by its capacity for on-board memory as well as multiplicative gates. KamilyaSmagulovaet al.[2019].

In a number of different applications, such as language modelling, speech-to-text transcription, and machine translation, amongst others, the LSTM network has been shown to have a significant impact.

Some readers in academic and industrial settings make the decision to learn about the Long Short-Term Memory network in order to evaluate the extent to which it can be applied to their own line of inquiry or to a specific scenario of practical use. The impressive benchmarks that were reported in the literature served as motivation for this decision. Sherstinsky, A. [2020].

Within the recurrent hidden layer of the LSTM are specialised units that are referred to as memory blocks. Memory cells are contained within the memory blocks. In the first iteration of the architecture, each memory block was equipped with both an input gate and an output gate. The input gate is responsible for controlling the flow of activations that are input into the memory cell. Controlling the flow of cell activations out into the rest of the network is the responsibility of the output gate. In later years, the memory block received an expansion in the form of the forget gate. Because the forget gate scales the internal state of the cell before adding it as input to the cell through the cell's self-recurrent connection, it is able to forget or reset the cell's memory in an adaptive manner.Figure 3.9 shows the architecture of an LSTM cell .

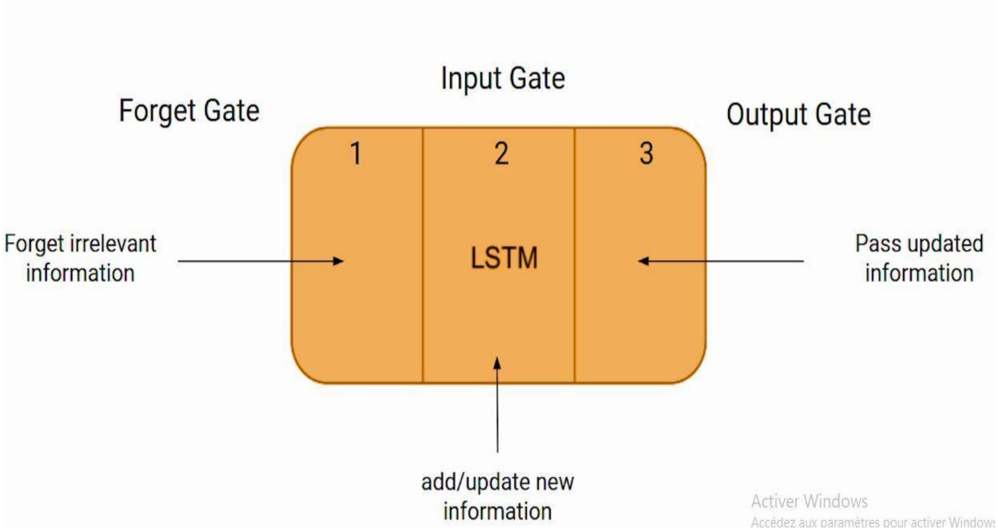

Figure 3.8 The architecture of an LSTM cell from ShipraSaxena[March 16, 2021]

CHAPTER III : DESIGN AND CONTRIBUTION

Because of the nature of the data that we have, we decided to use an LSTM architecture for our research.Because we're working with a succession of consecutive textual inputs, LSTM performs significantly better in this situation.It is potentially effective in language modelling since it may represent a phrase or a sequence of words.It is able to recall some of the information from the inputs it has received in the past and make predictions based on the results of more than one output. This architecture is appropriate for our model to employ in order to estimate the user's purpose based on the phrase that was supplied.

Pseudocode

Algorithm 1 LSTM prediction model pseudocode

```
1: data_preprocessing(inputs, targets)                ⇨Applying NLP methods to convert text data
2: input_train , targets_train , inputs_validation , targets_validation ←
Split_data (data,20)                   ⇨ Split data to training and validation sets (20% validation)
3: reshape_data(data)                   ⇨ Reshape data according to Embedding layer input
4:model ←Create Sequential Model()      ⇨ Creating and configuring the model
5: model.add_input_layer(input_dim,steps)
6:model.add_lstm(steps,dropout=0.2)
7:model.add_NN_layers(units, relu)
8:model.add_Dropout (0.5)
9:model.add_NN_layers(target_classes, softmax)
10:model.compile()
11: history←model.f i t(input s_train,targets_train, validation_data =
(inputs_validation,targets_validation),epochs = epochs)      ⇨ Training LSTM model

12: results ←model.evaluate(test_data)          ⇨ Testing the model from testing data
```

3.4 Architecture

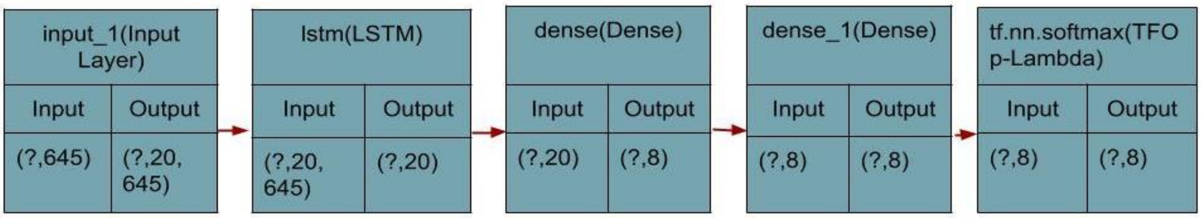

Figure 3.9 : LSTM model architecture and layers

There are five layers in our model, it starts with the initial layer which contains the initial data for the neural network. The LSTM layer will learn the long-term interconnections between the individual time steps. Then two Dense layers and a SoftMax layer that maps any given integer to either 0 or 1 as an output.

3.5Conclusion

Our system's design and algorithm were discussed in this chapter. There are two aspects to our system. Interaction is a significant component (mobile application). The second component is the python scripts for the backend server (translation, natural language processing, model prediction ...). The implementation and results will be disceussed in detail in the next chapter .

CHAPTER IV

CHAPTER IV: IMPLEMENTATION AND RESULTS

Chapter 4

Implementation and results

4.1 Introduction

We need to go through a development process that is carried out after a set of steps in order to construct an android app that satisfies the user's intentions. This will allow us to create an app that can be used on an android device.

Our project was presented and discussed in depth, including the theoretical aspects of the project and the methodologies employed.

In this chapter, we demonstrate the methodologies that we built by presenting some results. We provide the tools, platforms, and libraries that were used in the process of developing a chatbot for usage as an AirTravelassistant . In conclusion, we will review the supplied outcomes offered by our model and demonstrate the system interfaces we have developed.

4.2 Development tools and used platform

4.2.1 The Android application

React Native Software framework and NodeJS and many others extensible tools were used to build the Android app .

4.2.1.1 React Native

React Native is a javascript framework for writing real natively rendering mobile applications for iOS and Android. It’s based on React , Facebook’s JavaScript Library for building userinterfaces, but instead of targeting the browser it targets mobile platforms Eisenman (2016).

. React Native solved the problem of duplicating the codes and asymmetrical working of the apps.

React Native provides its users with one of a kind functionalities that boots efficiency while developing Android applications like :

. React components wrap existing native code and interact with native APIs via React's declarative UI paradigm and JavaScript.

. With the power of JavaScript, React Native lets you iterate at lightning speed.

. Cross-platform programming lets you construct one codebase for Android and iOS. Intelligente debuggingtools and errorreporting .

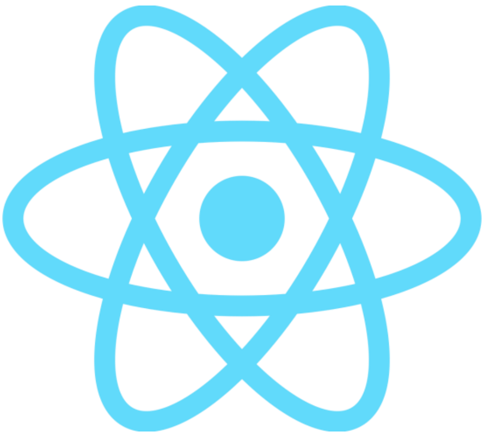

Figure 4.1 React Native

4.2.1.2 Java Script

Known for its dynamic and permissive nature , also its highly powerful language . JavaScript is a scripting or programming language that enables developers to implement complex features for Web browsers ,Web apps and game development, and lots more Nixon [2021] . Programmers use it for sophisticated apps. JavaScript has various peculiarities that are exploited by security and privacy threats. This is true when JavaScript has a familiar syntax but unusual semantics Guha [2010].

Figure 4.2 JavaScript

4.2.1.3 TypeScript

TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale [1] .Its headline feature is static typing, which makes working with JavaScript more predictable for programmers familiar with languages such as C# and Java Freeman [2019]. TypeScript adds additional syntax to JavaScript to support a tighter integration with your editor. Catch errors early in your editor [1] .Features of TypeScript can be applied selectively, which means you can use only those features useful for a specific project Freeman [2019].

Figure 4.3 TypeScript.

4.2.1.4 Expo

Expo is essentially a combination of tools and additional features wrapped around vanilla React Native and its goal is to build a JavaScript/TypeScript project that runs natively on all your users' devices [2] , also it enables rapid development without the need to spend much time setting up a development environment. It does not require setting up platform specific IDEs. Expo also ties junior developers to a certain version of React Native and the whole Expo infrastructure Tukiainen[2021] .

Figure 4.4 Expo.

4.2.2 Back-end processing

4.2.2.1 Google Colab

The Google research project known as Colaboratory, or "Colab" for short, was developed to assist students, data scientists, and artificial intelligence (AI) researchers with the teaching and study of machine learning. Google Colab is a free Jupyter notebook environment that requires no setup and runs entirely in the cloud. With Google Colab, it is possible to write and execute code, save and share our analyses, and access powerful computing resources, all for free from the browser Gunawan [2020] .Because the training of a deep learning model often requires a comprehensive CPU and GPU configuration, the Google Colab cloud platform is required for this activity.

4.2.2.2 Python

Python is currently the fastest growing programming language in the world, thanks to its ease-of-use, fast learning curve and its numerous high quality packages for data science and machine-learning Vallat [2018]. Python boosting both performance and productivity by enabling the use of low-level libraries and clean high-level APIs Raschka [2020] . As we outlined in the previous chapter, Python's extensive collection of useful libraries makes it the ideal platform on which to build our model for the Treatment process .

- **Pandas**

  Preparing CSV data for model training and analysis.

- **NLTK**

NLTK is a powerful Python package that provides a set of natural language algorithms. It is open-source, easy to use, has a large community, and a well-documented package. It consists of different algorithms such as tokenizing, part- of-speech tagging, stemming, sentiment analysis, and topic segmentation. NLTK helps the computer to analyze, pre- process, and understand the written text Khemani [2021] .

- **Tensorflow and Keras**

  Applied in the process of developing, training, and predicting outcomes in our deep learning model.

- **Matplotlib**

  Matplotlib is a Python plotting library which produces publication quality figures in a variety of formats and even interactive environments across platforms. As a Python library, Matplotlib can be used in scripts, the python and ipython shell and even web application servers Faes [2018] . It offers an object-oriented API for embedding plots in GUI applications.We use it to plot accuracy and loss graphs in our project.

4.2.2.3 APILibraries

- **Translation API**

  Rapid api is a platforme which contain a large API libraries .One of them is Deep Translation Library which provide acces to use translation libary without any restrictions.This makes a request to official Rapid API platform servers using this library .

- **Speech Hate API**

  The same platform mentiondprevisouly gives acces to use hate API library to filter the system from any innapropriateinput .

4.3 Chatbot Process

4.3.1 Establishing a connection to the back-end server:

```
* Serving Flask app 'main' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses (0.0.0.0)
  WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://127.0.0.1:5000
* Running on http://192.168.1.6:5000 (Press CTRL+C to quit)
```

Figure 4.5 Flask server

Launching the Flask server, which will be operating on the same network as the application, in order to handle the back end treatment .

4.3.2 User request :

It's at this point that the application establishes its connection with the flask server. The user opens the app, types his query into a text bar, and sends it to the airline's assistance with a click of the submit button; the app then responds with an answer. As shownis code below the main application interface.

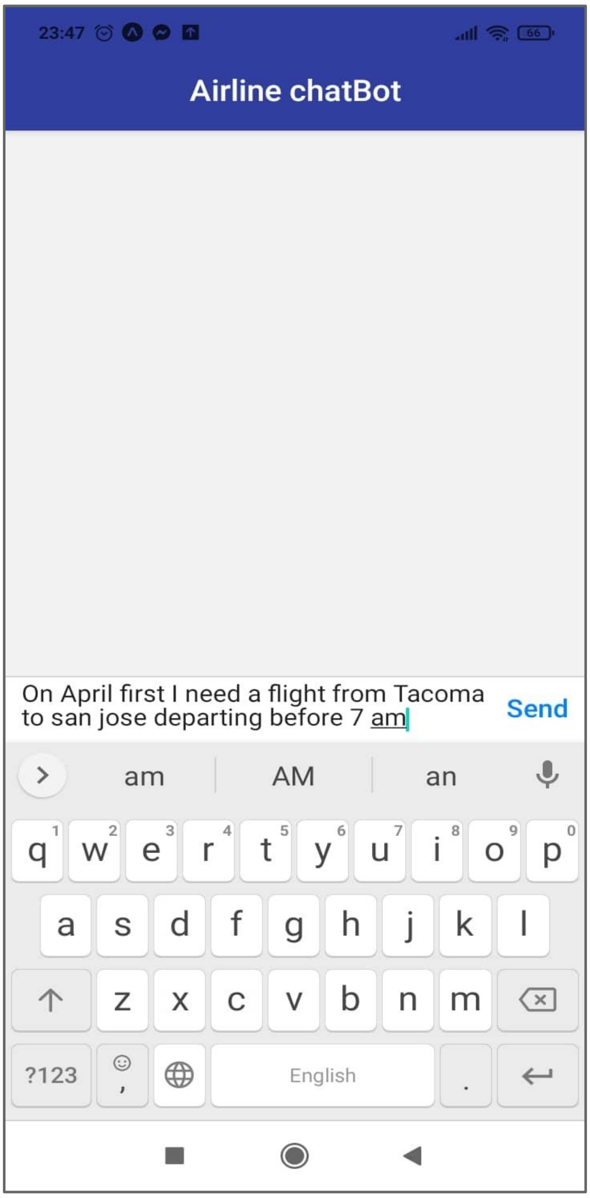

Figure 4.6 Application interface

4.3.3 Text translation :

The system of our chatbot normally works in english, if the received input it's not in english a translation must happen to english , the system uses three languages (English , Arabic ,French)

```
const detectLanguage = async (input: string) => {
  const options = {
    method: "POST",
    headers: {
      "X-RapidAPI-Key": "a5fe97e320mshbe04253044f8b5cp15279ejsne4358a1be896",
      "X-RapidAPI-Host": "deep-translate1.p.rapidapi.com",
    },
    body: `{"q": "${input}"}`,
  };

  const response = await fetch(
    "https://deep-translate1.p.rapidapi.com/language/translate/v2/detect",
    options
  );
  const jsonResponse = await response.json();
  const detectedLanguage = jsonResponse.data.detectedLanguage.language;
  return detectedLanguage;
};
export default detectLanguage;
```

Listing 4.1 : Detect user language

The used language is detected by the deepTranslatelibrary , and the translation happens using a Rapid API translation .

```
const translate = async (text: string, language = "en") => {
  const options = {
    method: "POST",
    headers: {
      "content-type": "application/json",
      "X-RapidAPI-Key": "a5fe97e320mshbe04253044f8b5cp15279ejsne4358a1be896",
      "X-RapidAPI-Host": "deep-translate1.p.rapidapi.com",
    },
    body: `{"q":"${text}","target":"en"}`,
  };

  const response = await fetch(
    "https://deep-translate1.p.rapidapi.com/language/translate/v2",
    options
  );
  const jsonResponse = await response.json();
  const translatedText = jsonResponse.data.translations.translatedText;
  return translatedText;
};

export default translate;
```

Listing 4.2 : Text translation

4.3.4 Code of conduct

Once the message is translated , a Speech hate API is applied to the input to verify and filter the input from any offensive language .

```
const axios = require("axios");

const isHateSpeech = async (text: string) => {
  const encodedParams = new URLSearchParams();
  encodedParams.append("text", text);

  const options = {
    method: "POST",
    url: "https://hate-speech-detection-for-user-generated-content.p.rapidapi.com/",
    headers: {
      "content-type": "application/x-www-form-urlencoded",
      "X-RapidAPI-Key": "a5fe97e320mshbe04253044f8b5cp15279ejsne4358a1be896",
      "X-RapidAPI-Host":
        "hate-speech-detection-for-user-generated-content.p.rapidapi.com",
    },
    data: encodedParams,
  };

  const response = await axios.request(options);
  const data = response.data;
  return data;
};

export default isHateSpeech;
```

Listing 4.3 : Code of conduct

4.3.5 Tokenization

The pseudocode below shows the Tokinzationmethod .

```
from nltk.tokenize import word_tokenize

text = "find a flight from memphis to tacoma dinner"
word_tokenize(text)
```

Listing 4.4 Tokenization code

In the previous chapter we already explained the tokenization process ,NLTK library Tokenizer is applied to the sentence input. Each token is compared to the model tokenize, and if the corresponding sequence exists, it is added to the category labed; otherwise, zero is added.

```
['find', 'a', 'flight', 'from', 'memphis', 'to', 'tacoma', 'dinner']
```

Figure 4.7 Output of tokenization method.

4.3.6 Stemming

The pseudocode below is applied to stem a sentence from our dataset .

```
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
pStemmer = PorterStemmer()
sentence = "find flights arriving new york  next saturday "
list_of_words = word_tokenize(sentence)
for words in list_of_words:
    print(words, " : ", pStemmer.stem(words))
```

Listing 4.5 Stemming pseudocode.

The previous chapters covered the concept of stemming in detail.This tokenized phrase will then be stemmed with the help of the Porter Stemmer technique from the NLTK library. As shown in the figure below 4.8 the stemming method outputs .

```
find  :  find
flights  :  flight
arriving  :  arriv
new  :  new
york  :  york
next  :  next
saturday  :  saturday
```

Figure 4.8 Output of stemming method

4.3.7 Model prediction :

Prediction of our model is performed as shown in the code below 4.6

```
#test on a single phrase
def predecter(model, phrase):
  predict_data = pd.DataFrame.from_dict({"text":[phrase]})
  predict_data["lower_text"]= predict_data.text.map(lambda x: x.lower())
  predict_data["tokenized"] = predict_data.lower_text.map(word_tokenize)
  predict_data["selected"] = predict_data.tokenized.map(lambda df: remove_stop(df, stop_punc))
  predict_data["stemmed"] = predict_data.selected.map(lambda xs: [stemmer.stem(x) for x in xs])
  predict_data["normalized"] = predict_data.stemmed.apply(normalize)
  tokenizer_predict = tokenizer.texts_to_sequences(predict_data.normalized)
  predict_padded = pad_sequences(tokenizer_predict, maxlen= 20, padding= "pre")
  x_predict_transformed= transform_x(predict_padded, tokenizer)
  y_encoder.inverse_transform(model.predict(x_predict_transformed))
  return y_encoder.inverse_transform(model.predict(x_predict_transformed))[0][0]

text = "on april first i need a ticket from tacoma to san jose departing before 7 am"
print( predecter(model,text))
```

Listing 4.6 : Prediction intent

Before applying a model prediction, we need to apply all the preprocessing steps to the sentence

from the input sequence "on april first i need a ticket from tacoma to sanjose departing before 7 am" , we get the intent as shown in figure 4.9

```
print( predecter(model,text))

atis_airfare
```

Figure4.9 Predicting intent output.

4.3.8 Answer retrieval :

The answer can be retrieved by sending a "POST" request from the android app to the "FLASK" server, which will be carrying the user request , and the server responds with an HTTP response containing the intent of the user.

```
from flask import Flask, request
import final_work
try:
    model = final_work.load_model("pretrained_model.h5")
except Exception:
    print("model n'existe pas")
app = Flask(__name__)
@app.route("/chat", methods=["GET", "POST"])
def chatBot():
    chatInput = request.form["chatInput"]
    return final_work.predecter(model, chatInput)
if __name__ == "__main__":
    app.run(host="0.0.0.0", debug=True)
```

Listing 4.7 API code of FLASK.

4.3.9 Output the answer :

The android application receives the user's intent and generates the response on the main layout as shown in the code below.

```
const getResponse = async (text: string) => {
  const language = await detect(text);
  const input = await translate(text);
  const isHateful = await isHateSpeech(input);
  if (isHateful) {
    return getHateSpeechResponse(language);
  }
  const response = await fetchResponse(input);
  const translatedResponse = await translate(response, language);
  return translatedResponse;
};

export default getResponse;
```

Listing 4.8 Receive the chatbot answer code

```
const fetchResponse = async (text: string) => {
  const response = await fetch("localhost:5000/chat", {
    method: "POST",
    body: JSON.stringify({ chatInput: text }),
  });
  const jsonResponse = await response.json();
  return jsonResponse;
};
```

Listing 4.9 Code of fetchResponsemethod .

4.4 System Interfaces and examples

In this section, we present some system usages and interfaces of our application with the three supported languages (Arabic - English - French)

• English user In this example in Figure 4.10 the Chatbot talks with an English person

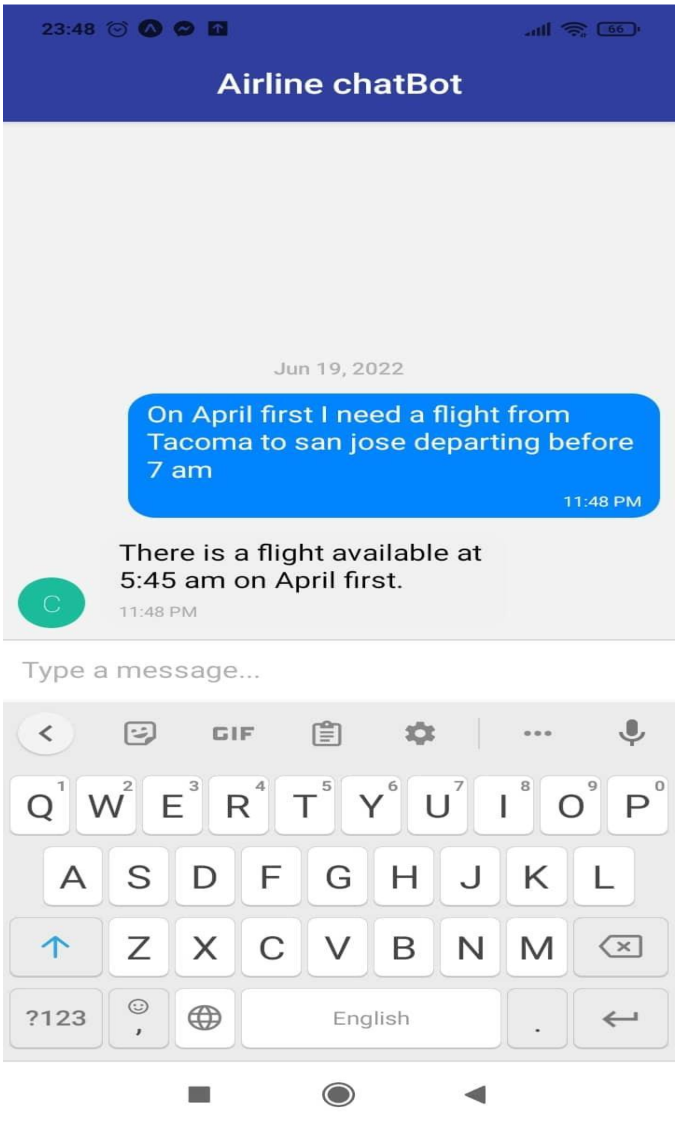

Figure 4.10 Chatbot test with English user.

• In this example in Figure 4.11the Chatbot talks to an Arabian person.

Figure 4.11 Chatbot answer with Arab user .

• French user In this example in Figure 4.12 the Chatbot talks to a French person.

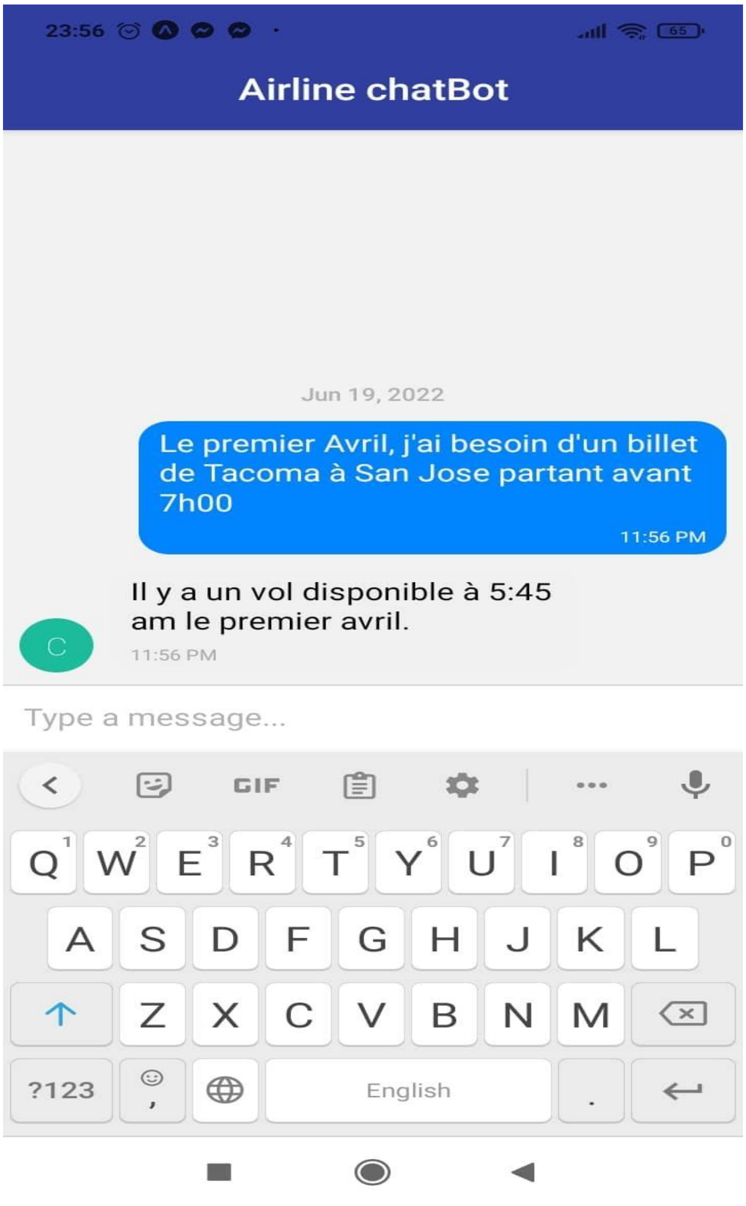

Figure 4.12 Chatbot answer with french user.

As shown in figure 4.13 the chatbot doesn't accept to communicate in case of hatespeech or any inappropriatemessage. Exactly it shows " Your message cannot be sent because it contains hate speech"

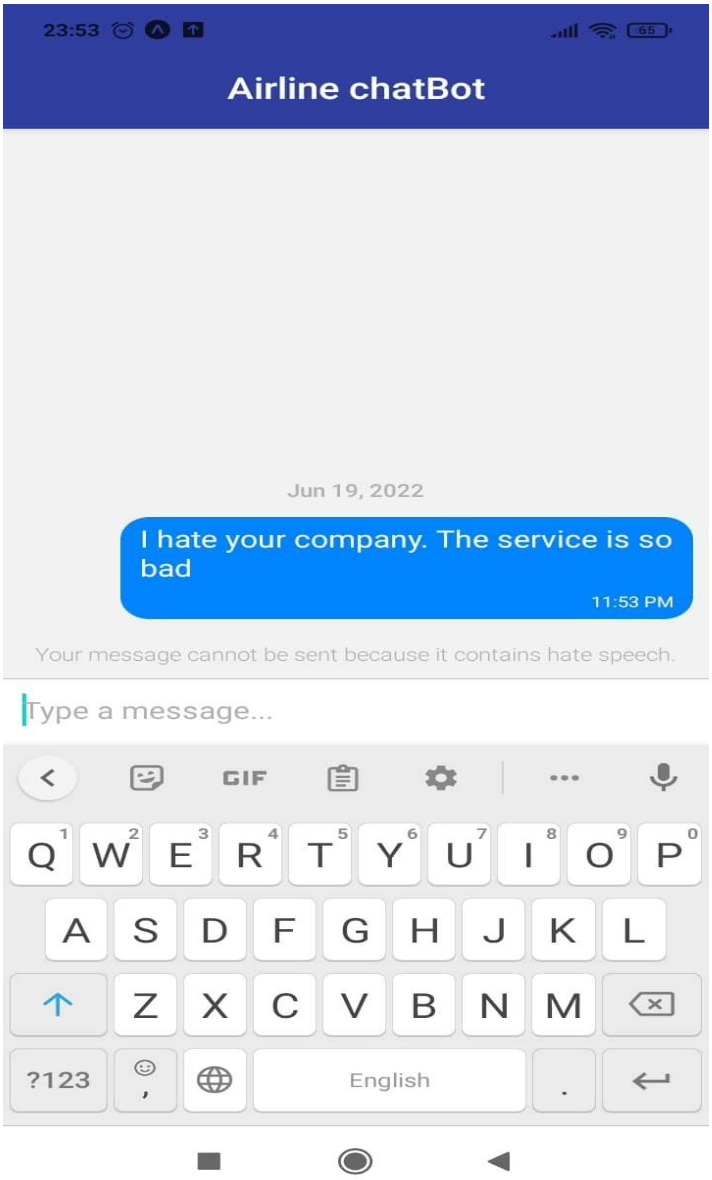

Figure 4.13 Chatbot answer in case of inappropriate message

.

4.5 Obtainedresults and discussion

The input dataset from HASSAN AMIN [2019] , grouped into a CSV file contains two columns target,text in a total of 8 different target (intent) for classification. The dataset was transformed into One-Hot encoding categorical targets by giving the value of One or Zero for each label.

4.5.1 Training

The training process of the model was in google colab. Training went for 70 epochs with the default batch size of 32. Here we applied two models BERT and LSTM model to learn the pattern between the question data and intents labels , in order to define and predict the intent correctly . When using BERT the accuracy was only 75% and the accuracy was low due to the imbalanced data .The code of LSTM model 4.10 shows the training process.

```
class LSTMModel(Model):

    def build_model(self, input_dim, output_shape, steps, dropout_rate, kernel_regularizer, bias_regularizer):
        input_layer= Input(shape= (steps, input_dim))

        #make lstm_layer
        lstm= LSTM(units= steps)(input_layer)
        dense_1= Dense(output_shape, kernel_initializer= he_uniform(),
                       bias_initializer= "zeros",
                       kernel_regularizer= l2(l= kernel_regularizer),
                       bias_regularizer= l2(l= bias_regularizer))(lstm)
        x= BatchNormalization()(dense_1)
        x= relu(x)
        x= Dropout(rate= dropout_rate)(x)
        o= Dense(output_shape, kernel_initializer= glorot_uniform(),
                 bias_initializer= "zeros",
                 kernel_regularizer= l2(l= kernel_regularizer),
                 bias_regularizer= l2(l= bias_regularizer))(dense_1)
        o= BatchNormalization()(o)
        output= softmax(o, axis= 1)

        loss= CC()
        metrics= AUC()
        optimizer= Adam()
        self.model= Model(inputs= [input_layer], outputs= [output])
        self.model.compile(optimizer= optimizer, loss= loss, metrics= [metrics])
```

Listing 4.10 Model Training architecture

Figure 4.14 Below shows the start of the training process. We can see at the beginning of the training,the accuracy started from 0.74 and the loss value was so high .

```
Epoch 1/70
124/124 [==============================] - 7s 21ms/step - loss: 7.1071 - auc: 0.7473 - val_loss: 5.4209 - val_auc: 0.8693
Epoch 2/70
124/124 [==============================] - 2s 17ms/step - loss: 3.9308 - auc: 0.9506 - val_loss: 3.3948 - val_auc: 0.9644
Epoch 3/70
124/124 [==============================] - 2s 17ms/step - loss: 2.3265 - auc: 0.9780 - val_loss: 2.2051 - val_auc: 0.9737
Epoch 4/70
124/124 [==============================] - 2s 15ms/step - loss: 1.5058 - auc: 0.9894 - val_loss: 1.6880 - val_auc: 0.9592
Epoch 5/70
124/124 [==============================] - 2s 15ms/step - loss: 1.0960 - auc: 0.9935 - val_loss: 1.2384 - val_auc: 0.9768
```

Figure 4.14 Beginning of the training with LSTM model

After 70 epochs,the training process accuracy reached 100% and validation accuracy 99%, and a lower loss value in the training (0.07) and (0.14) in validation loss, each epoch took about 15-17 seconds as shown in Figure 4.15.

```
Epoch 57/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0949 - auc: 1.0000 - val_loss: 0.1592 - val_auc: 0.9962
Epoch 58/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0900 - auc: 1.0000 - val_loss: 0.1634 - val_auc: 0.9963
Epoch 59/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0868 - auc: 1.0000 - val_loss: 0.1583 - val_auc: 0.9967
Epoch 60/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0856 - auc: 1.0000 - val_loss: 0.1573 - val_auc: 0.9967
Epoch 61/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0842 - auc: 1.0000 - val_loss: 0.1545 - val_auc: 0.9962
Epoch 62/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0818 - auc: 1.0000 - val_loss: 0.1539 - val_auc: 0.9962
Epoch 63/70
124/124 [==============================] - 2s 17ms/step - loss: 0.0819 - auc: 1.0000 - val_loss: 0.1514 - val_auc: 0.9958
Epoch 64/70
124/124 [==============================] - 2s 17ms/step - loss: 0.0800 - auc: 1.0000 - val_loss: 0.1553 - val_auc: 0.9966
Epoch 65/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0791 - auc: 1.0000 - val_loss: 0.1494 - val_auc: 0.9962
Epoch 66/70
124/124 [==============================] - 2s 17ms/step - loss: 0.0771 - auc: 1.0000 - val_loss: 0.1515 - val_auc: 0.9952
Epoch 67/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0748 - auc: 1.0000 - val_loss: 0.1524 - val_auc: 0.9965
Epoch 68/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0769 - auc: 1.0000 - val_loss: 0.1486 - val_auc: 0.9965
Epoch 69/70
124/124 [==============================] - 2s 15ms/step - loss: 0.0737 - auc: 1.0000 - val_loss: 0.1435 - val_auc: 0.9962
Epoch 70/70
124/124 [==============================] - 2s 16ms/step - loss: 0.0726 - auc: 1.0000 - val_loss: 0.1493 - val_auc: 0.9958
```

Figure 4.15 Ending of the training with LSTM model .

The training, which took around 15 minutes in total, produced excellent results for both accuracy and validation. The accuracynumberswereparticularlyimpressive.Seefigure 4.16.

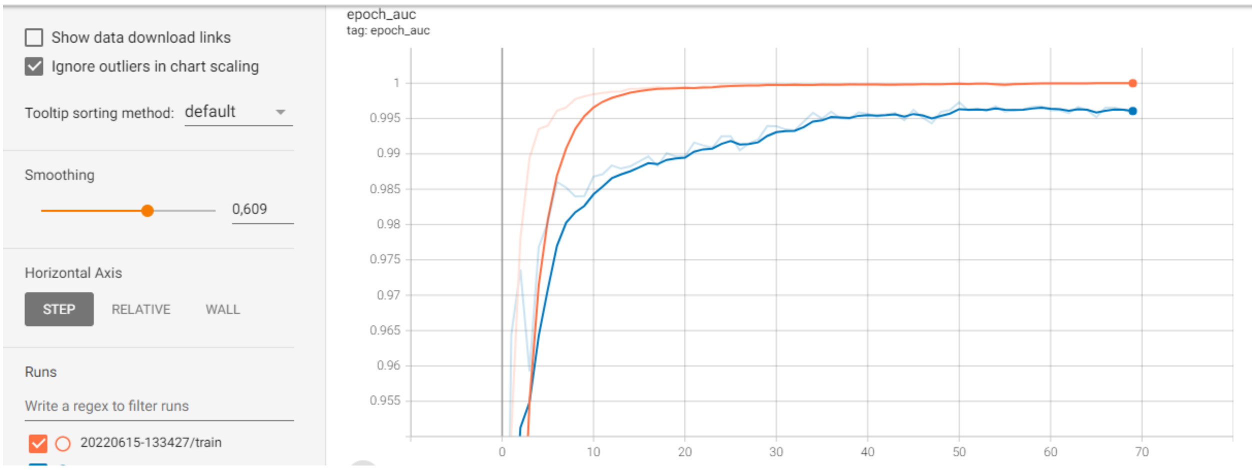

Figure 4.16 Training and validation accuracy

After conducting a couple of dozen tests and tweaking parameters, this model's accuracy went from 0.74 to 1, indicating that the LSTM model suited the data. The validation accuracy is also very good, reaching 0.99, since the proper intentions are predicted correctly most of the time.

We utilized CCE to calculate the loss. Classification's most prevalent function. Categorical Cross Entropy rises when projected probability diverges from label. The model minimizes the loss function, which summarizes all features of our approach into a single numerical result. LSTM seeks to learn from the data and minimize loss by employing back propagation while training progresses. After some epochs, the value reduces to 0.10 , indicating the model extracted features from the data set and identified the pattern to properly predict intentions.See figure 4.17 .

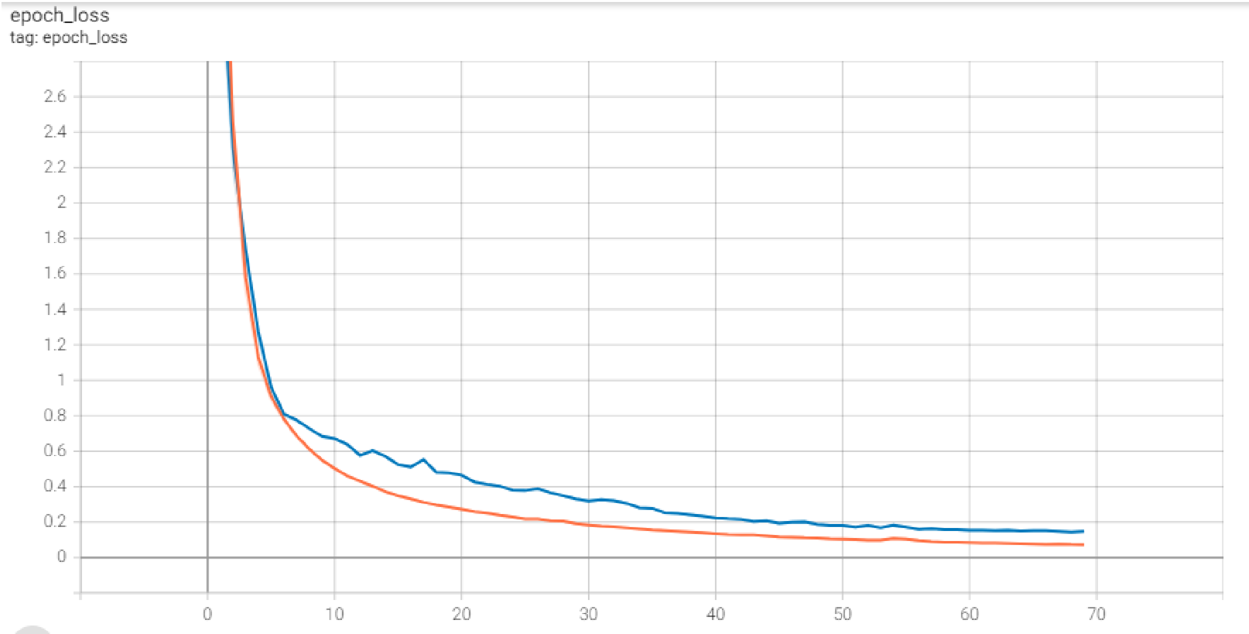

Figure 4.17 Training and validation Loss

4.6 Conclusion

In this chapter, we described both the implementation of our system and the outcomes that were acquired from doing so. Our selection of software applications and libraries was determined by the programming requirements at hand as well as the most effective means of accomplishing our objective. Our model showed a high accuracy in prediction, which represents the well-structured system and architecture, and the findings were encouraging overall.

GENERAL CONCLUSION

GENERAL CONCLUSION

Conclusion and Perspectives

The travel assistant is an essential part of the travel industry. Our purpose is to support the traveler and aid him as he requests. In this context, the suggested architecture based on Deep Learning and natural language processing methods has shown its usefulness in predicting the user's intents and producing a correct answer.

Long Short-Term Memory (LSTM) is used in our solution technique instead of traditional Artificial Neural Networks (ANNs) to recognize the user's intent, which is an improvement above previous works on other chatbots. Improved functionality is achieved by saving the user's input with the latest database changes. Our work uses a Dialogue state tracking is a dialog manager component responsible for the flow of the conversation between the user and the chatbot. It keeps a record of the interactions within one conversation in order to decide how to respond.

Perspectives

Our work may be expanded and developed to provide a better and more efficient travel assistant process. In the future, we want to include voice recognition into our system via the use of speech-to-text and text-to-speech techniques.The customer should not spend their time typing into the chatbot when they may be saving more time by speaking directly to it. One of the services that we want to add to our chatbot in the future is convenience and speed in making non-refundable hotel or travel reservations without having to pay a fee premium. To maximize the amount of people that utilize our platform, we should support many languages. It would be helpful if there was a method to make reservations and pay for them via the chatbot. The firms that can't handle customers from specific countries or regions should be kept under wraps .

References

Adamopoulou, E. and Moussiades, L., 2020, June. An overview of chatbot technology. In IFIP International Conference on Artificial Intelligence Applications and Innovations (pp. 373-383). Springer, Cham.

Applied Data Science in Tourism ,Roman Egger 2022.

A better way to evaluate Named Entity Recognition (NER) models.CharudattaManwatkar.Dec 12, 2020 .Available in : :https://towardsdatascience.com/a-pathbreaking-evaluation-technique-for-named-entity-recognition-ner-93da4406930c

Bhavesh Singh et al 2021 J. Phys.: Conf. Ser. 1831 012025 .A Trade-off between ML and DL Techniques in Natural Language Processing .

Ambition, word cloud .rakus .30 July 2017 available in :https://www.alamy.com/ambition-word-cloud-concept-on-white-background-image207176789.html.

Bhat, M. R., Kundroo, M. A., Tarray, T. A., & Agarwal, B. (2019). Deep LDA : A new way to topic model. Journal of Information and Optimization Sciences, 1–12.

Building a One Hot Encoding Layer with TensorFlow.George Novack. Jun 7, 2020 available in :https://towardsdatascience.com/building-a-one-hot-encoding-layer-with-tensorflow-f907d686bf39.

Bentaiba-Lagrid, M. B., Bouzar-Benlabiod, L., Rubin, S. H., Bouabana-Tebibel, thouraya, &Hanini, M. R. (2020). A Case-Based Reasoning System for Supervised Classification Problems in the Medical Field. Expert Systems with Applications, 113335.

B. V. Thazhathethil, U. Balasubramaniam and S. T. Abraham, "Coimbatore Destination Chatbot: A Study on Customer Preference," 2021 IoT Vertical and Topical Summit for Tourism, 2021, pp. 1-6.

[2] Basics of Expo . Available in :https://docs.expo.dev/index.html.

Bonnie Eisenman , 2016 .Learning React Native: Building Native Mobile Apps with JavaScript .O'Reilly Media .

Berka, P., 2020. Sentiment analysis using rule-based and case-based reasoning. Journal of Intelligent Information Systems, 55(1), pp.51-66.

ChrysovelidisGeorgios .(2019).Designing a Chatbot for Tourism .SCHOOL OF SCIENCE & TECHNOLOGY .International Hellenic University .

Create native apps for Android and iOS using React .Available in : **https://reactnative.dev/**

CODE OF CONDUCT ON MOBILE APP TRANSPARENCY ,NTIA May 16, 2013 Draft.

Dahouda, M. K., & Joe, I. (2021). A Deep-Learned Embedding Technique for Categorical Features Encoding. IEEE Access, 9, 114381–114391.

Dongsheng Wang.Semantic Representation and Inference for NLP.Computation and Language (cs.CL); Artificial Intelligence (cs.AI); Machine Learning (cs.LG).Semantic Representation and Inference for NLP.[Submitted on 15 Jun 2021].

Edwards-Stewart, A., Alexander, C., Armstrong, C.M., Hoyt, T. and O'Donohue, W., 2019. Mobile applications for client use: Ethical and legal considerations. Psychological Services, 16(2), p.281.

Fundamentals of Artificial Intelligence ,K.R. Chowdhary 2020 .

Francesco Colace, Massimo De Santo, Marco Lombardi, Francesco Pascale, AntonioPietrosanto , and Saverio Lemma .Chatbot for E-Learning: A Case of Study .International Journal of Mechanical Engineering and Robotics Research Vol. 7, No. 5, September 2018.

FADHLALLAH, G.M., 2021. A Deep Learning-based approach for Chatbot: medical assistance a case study.

Freeman, A., 2019. Essential TypeScript. Springer.

Faes, D.M., 2018. Use of Python programming language in astronomy and science. *arXiv preprint arXiv:1807.04806*.

Guha, A., Saftoiu, C., Krishnamurthi, S. (2010). The Essence of JavaScript. In: D'Hondt, T. (eds) ECOOP 2010 – Object-Oriented Programming. ECOOP 2010. Lecture Notes in Computer Science, vol 6183. Springer, Berlin, Heidelberg.

Grubic, N., Badovinac, S. and Johri, A.M., 2020. Student mental health in the midst of the COVID-19 pandemic: A call for further research and immediate solutions. International Journal of Social Psychiatry, 66(5), pp.517-518.

Gunawan, T.S., Ashraf, A., Riza, B.S., Haryanto, E.V., Rosnelly, R., Kartiwi, M. and Janin, Z., 2020. Development of video-based emotion recognition using deep learning with Google Colab. TELKOMNIKA, 18(5), pp.2463-2471.

Homem, T. P. D., Santos, P. E., Costa, A. H. R., da Costa Bianchi, R. A., & Lopez de Mantaras, R. (2020). Qualitative Case-Based Reasoning and Learning. Artificial Intelligence, 103258.

Introduction to natural language processing R. Kibble CO3354 2013.

Introduction to Long Short Term Memory (LSTM).Shipra Saxena — March 16, 2021.Available in https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/

Jin, Y. (2017). Development of Word Cloud Generator Software Based on Python. Procedia Engineering, 174, 788–792.

Jaideepsinh K. Raulji ,Jatinderkumar R. Saini, Stop-Word Removal Algorithm and its Implementation for Sanskrit Language 2016.

Khemani, B. and Adgaonkar, A., 2021, April. A Review on Reddit News Headlines with NLTK tool. In *Proceedings of the International Conference on Innovative Computing & Communication (ICICC)*.

Klaus, P. and Zaichkowsky, J.L., 2022. The convenience of shopping via voice AI: Introducing AIDM. Journal of Retailing and Consumer Services, 65, p.102490.

Li Deng • Yang Liu ,Deep Learning in Natural Language Processing 2018.

Li, J., Sun, A., Han, J. and Li, C., 2020. A survey on deep learning for named entity recognition. IEEE Transactions on Knowledge and Data Engineering, 34(1), pp.50-70.

Leveraging capabilities of NLP in manufacturing industries.Kaviraju. October 17, 2021 . available in :https://industry40.co.in/leveraging-capabilities-of-nlp-in-manufacturing-industries/

Li, H., Zhu, Y., Burnside, E.S., Drukker, K., Hoadley, K.A., Fan, C., Conzen, S.D., Whitman, G.J., Sutton, E.J., Net, J.M. and Ganott, M., 2016. MR imaging radiomics signatures for predicting the risk of breast cancer recurrence as given by research versions of MammaPrint, Oncotype DX, and PAM50 gene assays. Radiology, 281(2), p.382.

Lawson, D.A., Bhakta, N.R., Kessenbrock, K., Prummel, K.D., Yu, Y., Takai, K., Zhou, A., Eyob, H., Balakrishnan, S., Wang, C.Y. and Yaswen, P., 2015. Single-cell analysis reveals a stem-cell program in human metastatic breast cancer cells. Nature, 526(7571), pp.131-135.

Liu, W., Tan, R., Cao, G., Yu, F. and Li, H., 2020. Creative design through knowledge clustering and case-based reasoning. Engineering with Computers, 36(2), pp.527-541.

Murad, C., Munteanu, C., Clark, L., & Cowan, B. R. (2018). Design guidelines for hands-free speech interaction. Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct - MobileHCI ’18.

Mani, V., Agrawal, R. and Sharma, V., 2014. Supplier selection using social sustainability: AHP based approach in India. International Strategic Management Review, 2(2), pp.98-112.

Machine-Learning is Leading the Self-Imporving Help Desk : Case-Based Reasoning (CBR) systems .2021.Avalaible in :https://www.givainc.com/blog/index.cfm/2021/7/8/case-based-reasoning-cbr-meaning-ai-help-desk

Manaswi, N.K., Manaswi, N.K. and John, S., 2018. Deep learning with applications using python (pp. 31-43). Bangalore, India: Apress.

Natural Language Processing: State of The Art, Current Trends and Challenges Diksha Khurana, Aditya Koli, Kiran Khatter, Sukhdev Singh 2017 .

Nixon, R., 2021. Learning PHP, MySQL & JavaScript. " O'Reilly Media, Inc.".

Ngai, E.W., Lee, M.C., Luo, M., Chan, P.S. and Liang, T., 2021. An intelligent knowledge-based chatbot for customer service. Electronic Commerce Research and Applications, 50, p.101098.

Okonkwo, C.W. and Ade-Ibijola, A., 2021. Chatbots applications in education: A systematic review. Computers and Education: Artificial Intelligence, 2, p.100033.

Palanica, A., Flaschner, P., Thommandram, A., Li, M. and Fossat, Y., 2019. Physicians' perceptions of chatbots in health care: cross-sectional web-based survey. Journal of medical Internet research, 21(4), p.e12887.

Reshamwala, A., Mishra, D. and Pawar, P., 2013. Review on natural language processing. IRACST Engineering Science and Technology: An International Journal (ESTIJ), 3(1), pp.113-116.

Ronald Leenes • Rosamunde van Brakel Serge Gutwirth • Paul De Hert . Data Protection and Privacy: (In)Visibilities and Infrastructures 2017 .

Ren, F., & Bao, Y. (2019). A Review on Human-Computer Interaction and Intelligent Robots. International Journal of Information Technology & Decision Making.

Raschka, S., Patterson, J., & Nolet, C. (2020). Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence. Information, 11(4), 193.

Santra, D., Mandal, J.K., Basu, S.K. and Goswami, S., 2020. Medical expert system for low back pain management: design issues and conflict resolution with Bayesian network. Medical & Biological Engineering & Computing, 58(11), pp.2737-2756.

Sun, X., Sun, J., Li, T., Zheng, S., Wang, C., Tan, W., Zhang, J., Liu, C., Ma, T., Qi, Z. and Liu, C., 2019. Flexible tactile electronic skin sensor with 3D force detection based on porous CNTs/PDMS nanocomposites. Nano-micro letters, 11(1), pp.1-14.

Samuel Manoharan. A smart image processing algorithm for text recognition information extraction and vocalization for the visually challenged. Journal of Innovative Image Processing (JIIP), 1(01):31–38, 2019.

Shuai-Shi Liu, Tian, Y.-T., & Dong Li. (2009). New research advances of facial expression recognition. 2009 International Conference on Machine Learning and Cybernetics.

Smagulova, K., & James, A. P. (2019). A survey on LSTM memristive neural network architectures and applications. The European Physical Journal Special Topics, 228(10), 2313–2324.

Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D: Nonlinear Phenomena, 132306.

[1] Should You Use TypeScript?.2019 .Available in : https://www.typescriptlang.org/ .

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M. Smith, Y-Lan Boureau, Jason Weston. 2020.Recipes for building an open-domain chatbot.

Tokenization in NLP.SATISH GUNJAL (2019).Available in :https://www.kaggle.com/code/satishgunjal/tokenization-in-nlp/notebook.

T. K. Shivaprasad and J. Shetty, "Sentiment analysis of product reviews: A review," 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT) .

Thangaprakash Sengodan • M. Murugappan • Sanjay Misra .Advances in Electrical and Computer Technologies Select Proceedings of ICAECT 2019.

Thorat, S.A. and Jadhav, V., 2020, April. A review on implementation issues of rule-based chatbot systems. In Proceedings of the International Conference on Innovative Computing & Communications (ICICC).

Thakkar, M.D., Sanghavi, C.U., Shah, M.N. and Jain, N., 2021, March. Infini–A Keyword Recognition Chatbot. In 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) (pp. 1036-1042). IEEE.

The essential Guide to Neural Network Architectures.PragatiBaheti.May 26.2022 .Available in :https://www.v7labs.com/blog/neural-network-architectures-guide.

.

Tukiainen, J., 2021. Integrating React Native Application Into Third-Party mHealth Application.

Tran, A.D., Pallant, J.I. and Johnson, L.W., 2021. Exploring the impact of chatbots on consumer sentiment and expectations in retail. Journal of Retailing and Consumer Services, 63, p.102718.

Vallat, R., 2018. Pingouin: statistics in Python. J. Open Source Softw., 3(31), p.1026.

What are Stop words? .18 May, 2022 available in :https://www.geeksforgeeks.org/removing-stop-words-nltk-python/

Young Jin Kim, Hany Hassan Awadalla .FastFormers: Highly Efficient Transformer Models for Natural Language Understanding [Submitted on 26 Oct 2020].

Youness Tabii • Mohamed Lazaar Mohammed Al Achhab • NourddineEnneya (Eds.) Big Data, Cloud and Applications 2018 .

Yu, Y., Si, X., Hu, C. and Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. Neural computation, 31(7), pp.1235-1270.

Zumstein, D., & Hundertmark, S. (2017). Chatbots-An interactive technology for personalized communication, transactions and services. IADIS International Journal on WWW/Internet, 15(1), 96-109.